Politecnico di Torino

Master's Degree in Environmental and Land Engineering

Politecnico di Torino

Master's Degree Thesis

Parameterization of a Regional Hydrologic Model for Piedmont: simulation of large-scale floods

Supervisor:
Prof. Alberto VIGLIONE

Candidate:
Luca LOMBARDO

OCTOBER 2022

ACKNOWLEDGMENTS

For the realization of this master thesis, I want to thank in first place my supervisor, Professor Alberto Viglione, for the great help and continuous interest and attention throughout the entire duration of the academic year. A further thank is due to Dr. Matteo Pesce who supported me whenever I needed it by always finding a moment to clear up any possible doubt.

I then want to thank my family who has always been supportive during these long years at the university despite the various ups and downs that occurred along the way.

A thought then goes to my classmates which have made the long hours of lessons and exams more bearable in a period as particular and unexpected as the last two years have been for all of us.

Last but not least, a special thanks to my dear friends Alessia and Nike who, both when close and when far away, have backed me up and always supported me. Sharing this moment with you simply adds that little extra to this special day.

ABSTRACT

The aim of this study is the regionalization of the parameters of a hydrological model (***TUWmodel***) for the Piedmont region, through an innovative procedure involving machine learning techniques (***HydroPASS***) for estimation of runoff at ungauged locations. The regionalization is performed by the determination of a regional functional relationship between model parameters and a selected set of descriptors (*CDs*) over a reference grid (resolution around $12\ km$) without any a priori assumptions about the function itself. The first part of the study involved the determination of the needed input data to then move to the actual application of the algorithm. The climate data (Precipitation, mean temperature, and *PET*) over the grid (from 1961 to 2021) have been obtained from *ARPA Piemonte* and simple calculations just like the discharge observations at 127 gauged stations. This information has been used to perform the local lumped calibrations of the *TUWmodel* parameters for each catchment (***DE algorithm***), resulting in an average **Kling-Gupta efficiency (*ME*) of $\mathbf{0.86}$**. The 52 selected descriptors, evaluated for both the catchments and the reference grid, are instead divided into to five groups (climate, morphology, land use, soil, and curve numbers). After the application of the regionalization algorithm, the searched distributed parameters have been obtained resulting in a **distributed Kling-Gupta efficiency (*ME*) of $\mathbf{0.7}$**. Further analysis shown the flexibility of the regionalization procedure when provided with more limited number of *CDs*, resulting in similar, not degraded, efficiencies. The newfound distributed parameters have then been applied for the reconstruction of five historical flooding events (1994, 2000, 2008, 2016, 2020) over the Piedmont region. The model, despite presenting some limits in the quality of the observed data discussed in the thesis, performed reasonably well resulting in coherent simulations, both spatially and temporary, at ungauged locations, in agreement with the descriptions contained in the official reports redacted by the local authorities.

TABLE OF CONTENTS

INDEX OF FIGURES AND TABLES

ATTACHMENTS

1. INTRODUCTION

1.1 Continuous discharge estimation and regionalization

"Hydrology is the branch of science concerned with the properties of the earth's water, and especially its movement in relation to land". Part of this science is the continuous streamflow estimation. Streamflow (i.e., the amount a water flowing, in a precise cross section, in a channel or river) is monitored and observed at gauged locations but can only be estimated, reconstructed, or forecasted in ungauged ones. Discharge timeseries are of fundamental importance in many fields of practice, ranging from ecology to engineering: hydrological signatures derived by such data are for example used in the design of all critical infrastructure (reservoirs, drainage systems) to be able to sustain events with different return periods, just like for the definition of legislative frameworks to prevent and face damages caused by flooding events, to name a few. Despite the necessity of such information, most rivers and stream reaches and tributaries in the world are ungauged or poorly gauged (Sivapalan et al. 2003; Young 2006; Mishra and Coulibaly 2009). An ungauged or poorly gauged basins are, according to Sivapalan et al. (2003), the ones with inadequate records, both in terms of quality and quantity with respect to a variable of interest (the same concept can be applied outside the strict hydrology, like for temperature measurements). In this context, of fundamental importance is the possibility and capability of reconstruct, starting from the available observation at gauged location, streamflow timeseries where there is not a monitoring network (*PUB – Prediction Ungauged Basins, see "Runoff Prediction in Ungauged Basins", Gunter Bloschl, 2013*). Currently reconstructions and forecasts are mainly realized relying on the utilization of conceptual and semi-distributed models (Tara Razavi et al., 2013). A conceptual model reproduces the hydrological processes in a simplified way, schematizing the processes without directly solving physical equations: each model is characterized by several parameters that can be tuned (calibrated) to modify the resulting output. A full classification and description of the possible models will be provided in chapter 4: for the moment, our interest is to review the different historical approaches to the regionalization process and then introduce the innovative methodology on which this entire work is based on.

Blöschl and Sivapalan (1995) defined as regionalization the process of transferring the information from one catchment to another, a procedure that can provide good results if the catchments are in some sense similar. The transferred information can be of different nature; in our case it will be the model parameters. Indirectly this definition implies that the regionalization process relies on a very strict assumption: the gauged and ungauged catchments should be in a homogeneous region. In other words, it is assumed that similar conditions (climate, topography, land use, etc…) will generate a similar response no matter where we are inside the area of study. Due to the lack of observations

(used to calibrate the model in the gauged location), a regionalization procedure is universally recognized as a challenge which outcomes success are often limited.

1.2 Regionalization approaches

A general and simplified form of regional model can be the following, as proposed by Wagener and Wheater (2006):

$$\hat{\theta}_L = H_R(\vartheta_R|\Phi) + v_R \quad (eq.1.1)$$

Where $\hat{\theta}_L$ is the hydrological variable of interest at the ungauged site, H_R is the regional functional relationship for the same variable, ϑ_R are sets of the variable of interest at gauged location and Φ a set of catchment attributes. v_R is the error term. In other words, a function between model parameters and attributes (in this work they will be called catchment descriptors, *CDs*) is derived starting from the gauged catchments (where parameters can be calibrated). The same found relation is then applied where only attributes are available to predict parameters for the reconstruction of discharge data at ungauged sites.

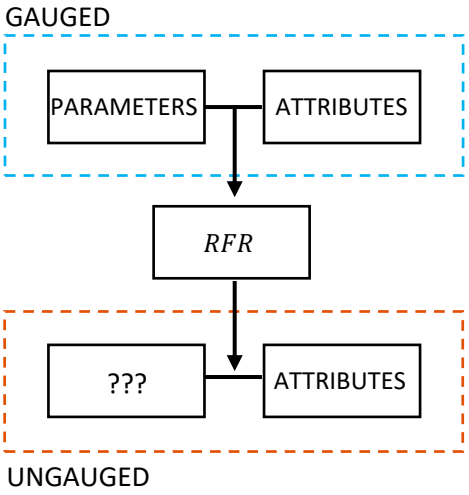


Figure 1 - "Conceptual scheme of regionalization, in which regional functional relationships (RFR) are sought to derive hydrological model parameters for ungauged catchments from those estimated in gauged catchments"

Five general steps can be identified for the definition of a regional functional relationship (from now on RFR):

- Definition at the gauged location of the hydrological variable (or variables) to be regionalized. In case of model parameters, a calibration must be conducted (see chapter 4).
- Collection and definition of the catchment descriptors (attributes) at gauged and ungauged sites: not always the data needed for their computation, generally in form of indices or shares (like for land use), are available or easily accessible. Moreover, the discussion about which *CDs* and in which number are needed for a good regionalization is still an open question without a clear answer. Often the selection is initially guided by intuition or by the reasoning about the main hydrological processes at the base of runoff generation.
- Development of the RFR: here different approaches can be used to obtain the function needed to connect hydrological variable and descriptors (see chapter 6).
- Run a validation procedure: mathematically a RFR will always be found after the first three steps, but such function may still not be suitable for the regionalization applications. Before applying the regional model, the relation is tested in different ways: the most used method is the *one-out cross* validation where each basin is, in turn, considered fictionally ungauged to obtain a simulated flow, but still comparable with observed data with an objective function. If the validation procedure gives acceptable results, the regional model can be applied to real ungauged catchments.
- Perform an uncertainty analysis about the obtained hydrological variables at ungauged locations.

The third step is where the major differences can be introduced from a method to another, changing the actual strategy used to determine the regional function relationship:

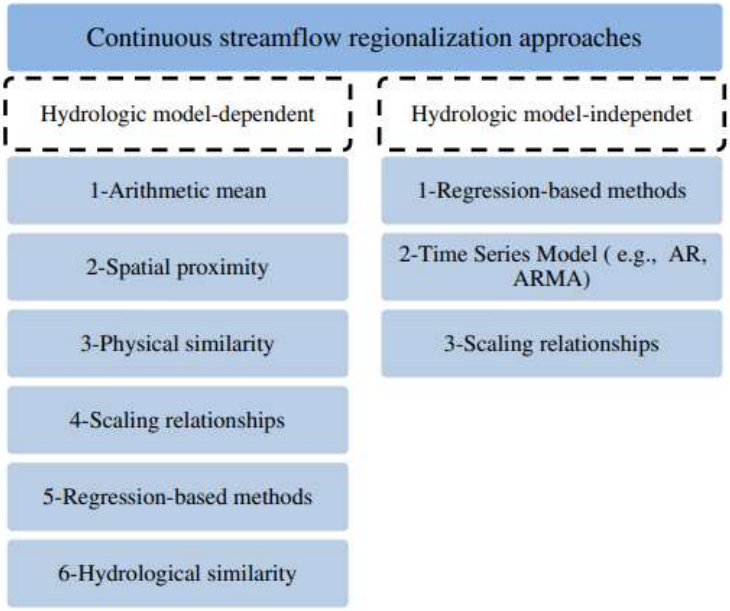


Figure 2 - Schematic of two main classes and subdivisions of continuous streamflow regionalization methods (Tara Razavi et al., 2013)

Inside the model-dependent approaches, the first, and simpler, to have been used are the *arithmetic mean* and the *spatial proximity*. Both do not directly require the definition of catchment descriptors, transferring the model parameters by mean of simple averages. In *arithmetic mean* the parameters at gauged location are averaged and the mean is used at the ungauged sites. The mean can be done globally on all the available basins or locally considering just the surrounding catchments. In *spatial proximity* is instead utilized an interpolation technique that is function of the geographic location of the ungauged with respect to the gauged catchments (like a weight to give more importance to the closest basins). Kriging is by far the most used and successful method of this kind.

A third class of approaches that still does not directly use the descriptors are the *scaling relationships*. The variables are scaled to sub-catchment areas assuming that the streamflow contribution from each sub-catchment to the total catchment yield is proportional to a ratio of the catchment area or other attributes (Schreider et al., 2002).

Physical similarity approaches subdivide gauged and ungauged catchments in sub-groups depending on their descriptors. Basins with similar descriptors are grouped together (the similarity is determined with multivariate statistics analysis). Then, for each group, starting from the hydrologic variable at the gauged locations, a regional relationship (for example again with a simple arithmetic mean) is found and applied to all ungauged sites of the same group.

More sophisticated approaches are instead the *regression methods*: linear or non-linear relations between descriptors and parameters can be used. The main idea is to select a priori which *CDs* are more important and the structure of the RFR. Using the available information at gauged locations such relation is then tuned to give the best performance. At the same time, as already stated, identifying which descriptor is more relevant than another is difficult, and more than one set of descriptors can be combined to well represent the observations. Regression methods are indeed a powerful tool but limited by their intrinsic fixed structure already decided before the regionalization even starts.

Despite different methods can result in different outcomes, in different regions, all have been used with discrete success. Inside this framework, in 2020, Merz at al. in their article *"Parameters's control of distributed catchment models; how much information is in conventional catchment descriptors?"* proposed a new innovative approach to tackle the problem of regionalization. Their proposed algorithm, ***HydroPASS (Parameter Set Shuffling),*** follows the idea of regression methods, but do not require a priori assumptions about the *CDs* used and about the relationship between model parameters and catchment descriptors. On the contrary such relation, and so the structure of the RFR are derived by the same algorithm during the regionalization by mean of machine learning techniques. The great advantage of this new methodology is the fact that it does not rely on choices just made based on experience or intuition. Such decisions may in fact reveal wrong: catchment descriptors relative importance can change with scale and region and so is difficult to find a single structure always working. Moreover, in literature (like Bloschl et al., 2013) there are several studies that report that in regionalization studies, unambiguous relationship, even in the same area, cannot be found. A full description of the algorithm will be provided in chapter 6; for the moment, in the next paragraph the goal of this study and the framework adopted to achieve it are detailed.

1.3 Scope and operational framework

In the following study the objective is the application of the regionalization procedure proposed by Merz et al. on the Piedmont region, in the north-west part of Italy, starting from the available discharge data at some gauged locations and from open-source databases for the calculation of the catchment descriptors. The obtained regional functional relationship will then be applied on a grid of *HUs* (hydrological units) to simulate historical floods that struck the area in the past and to obtain a spatially detailed reconstruction of the river discharges during those events.

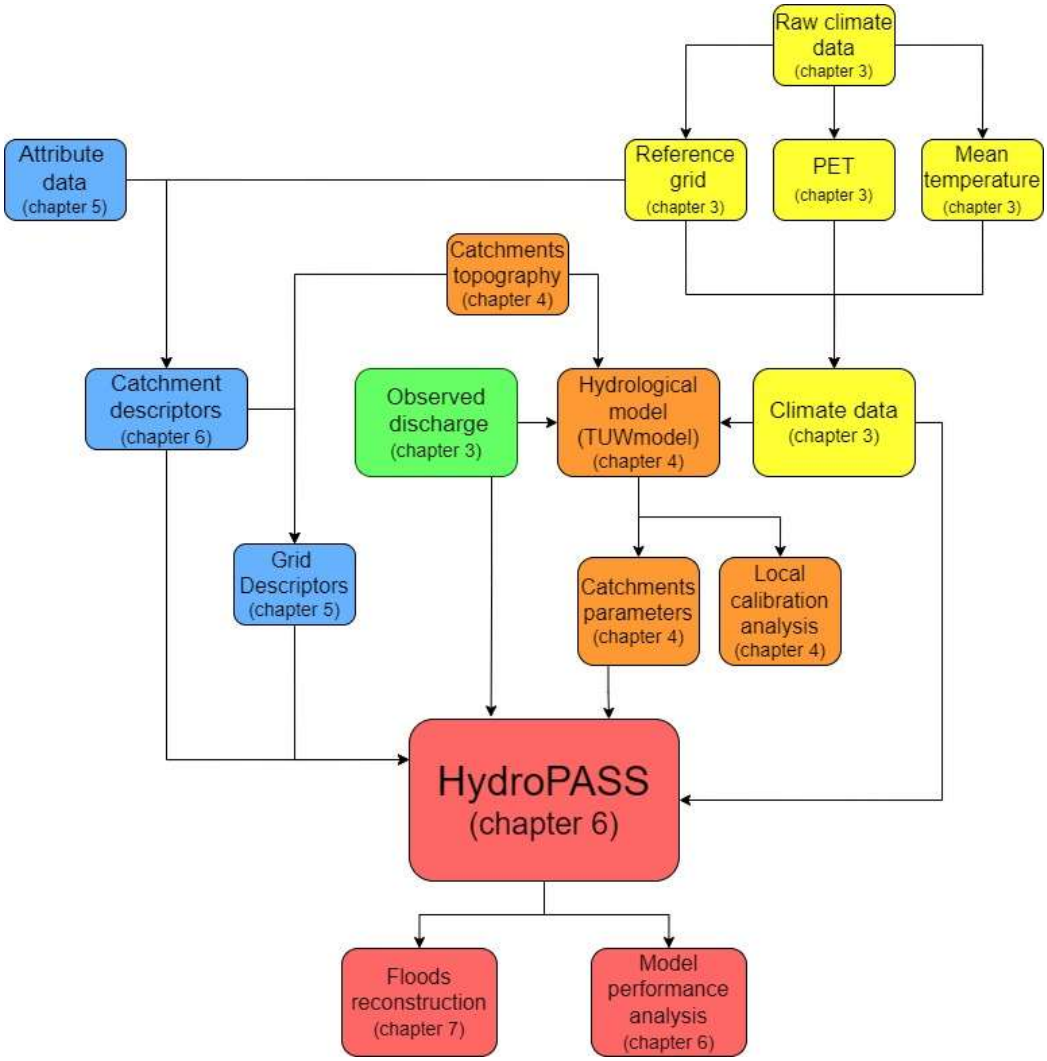


Figure 3 - Flow Chart with sections of the study. Raw climatological data and manipulation (yellow, chapter 3), Observed discharge data (green, chapter 3), local lumped calibration (orange, chapter 4), descriptors determination (light blue, chapter 5), and HydroPASS application (red, chapter 6)

The proposed flow chart (Figure 3) follows the general regionalization procedure already described in the previous paragraph. Some elements are interconnected and are necessary both directly inside the regionalization routine and for the definition of other required *HydroPASS* input elements. The colors have been used as far as possible to group together single aspects of the study as they will be described in detail in the following chapters:

a) A first section (in yellow) is dedicated to the manipulation of raw climatological data to obtain the required climate input in the desired format. In this section is also determined the reference grid over which most of the parameter and indices of the entire work will be referred to (Chapter 3).

b) The second section, composed by the single green box, is about the observed discharge timeseries over the gauged catchments. No heavy manipulation where needed on them (Chapter 3).
c) The following section (orange) is the one related to the determination of the local lumped parameters for the gauged catchments for the regionalization. Apart from the obtainment of the parameter sets, inside there is also a dedicated box about the analysis of the quality of such calibrations (efficiency by class, capability of reconstructing hydrological signatures, parameters uncertainty and equifinality) (Chapter 4).
d) The fourth section (light blue) contains all the calculations for the catchment descriptors utilized. Despite the name, here are calculated the descriptors also for each *HU* of the distributed regional model (Chapter 5).
e) The last section (red) is the application of the algorithm with machine learning techniques, the analysis of the obtained outcomes and the application of the distributed regional model for floods reconstruction over the Piedmont region (Chapters 6 and 7).

At the beginning of each chapter (or whenever needed) a cropped and compacted version of this flow chart will be reported to recall where we are and focus the attention on the steps currently discussed for a better comprehension.

Before starting with the description of the procedure, still follows a short second introductory chapter about the main software and environment utilized for all the computations reported afterward. On few occasions, other software has been used for some individual operations; they will not have a dedicated introduction but will be opportunely cited whenever they have been utilized.

2. THE R ENVIRONMENT

2.1 General description

The entire work and results provided, starting from chapter 3, have been obtained inside the *R* environment. *R* is a language (production of codes and scripts through the implementation of functions and commands) implemented in the first 1990's by Robert Gentleman and Ross Ihaka at the University of Auckland. This language is strictly related to a previous language called *S* for Statistical Computing, invented by John Chambers, Rick Becker, and others at Bell Labs in the mid-1970s and made available later in the 1980s. Since the beginning of the development of the *R* project in 1995 it was decided to establish it as a free and open-source software, without any need for a particular license for its use and results publication. After a first period of development under the work of the *R* core group (a group of core members having access to the source code) a first public release was made available in February 2000. Despite many researchers refer to *R* as a statistical system, the developers in the official *"Introduction to R"* prefer the definition of an environment *"within which many classical and modern statistical techniques have been implemented"*. The term *"environment"* is intended to characterize it as a fully planned and coherent system, rather than an incremental accretion of very specific and inflexible tools. In fact, *R* is famous not only for the statistical analysis it can support, but also for the graphical tools made available for the realization of quality plots. Base *R* includes an effective data handling and storage facility, operators for calculation on arrays (in particular, matrices), a collection of intermediate tools for data analysis, graphical facilities for data display and the programming language itself for the implementation of conditionals, loops, and cycles.

One, if not the most, powerful advantages of using *R*, is the extended possibility of expanding the available functions and tools inside the software with the installation of external packages: a package is nothing more than a collection of additional functions and/or functionalities as developed by independent authors who decided to spontaneously contribute to the growth of the open-source software. Also, the base functions contained in base *R* are stored inside the so-called *standard packages*, automatically loaded when the software is booted. Another set of packages (the *recommended packages*) are also already present in base *R*, but not automatically loaded. Any additional package must instead be first downloaded and installed from one of the official sources with the function *"install.packages"* (the biggest repository is *CRAN – Comprehensive R archive network*) and then loaded for the use with the dedicated function *"library"*.

The realization of an *R* package must follow some quite strict rules to meet the needed requirements for a fast and easy usability by the users. In its more limited form, a package (that in practice is a directory) contains the actual code for the added functions (inside a directory named *R*), the description of the package (with information like name, version, authors, and other metadata) inside a plain text file with no extension and a brief description of the function included (that can be visualized with the *"help"* command in the software). Other additional elements can be present (and must be present to be compliant with the requirements to be published in *CRAN*) like example data or indications to required or suggested complementary packages, but they will not be here described in detail. A full description for the creation of a complete package can be found in the *"Writing R extension"* official guidelines.

2.2 R Studio

It must be noticed that *R* is a language and as such does not possess per se a graphical interface (working from line command as the *LINUX* operative system). Being the calculation mostly been conducted on a device using the Windows operation system, an additional software has been used. *R Studio* is an integrated development environment for *R* that adds a full graphical interface for an easier and more direct use (not different from other software like *MATLAB* or *Octave*). In details it introduces a console to write and modify the code, a syntax-highlighting editor plus tools for plotting (with a direct visualization), debugging and workspace management.

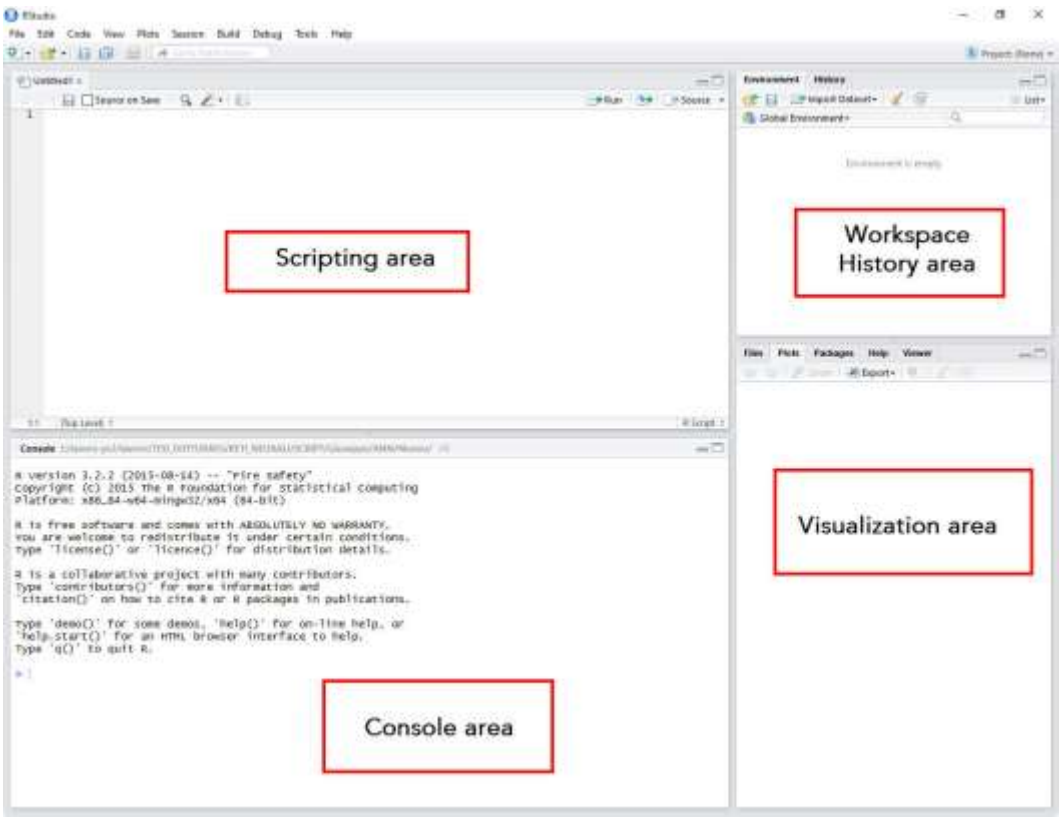
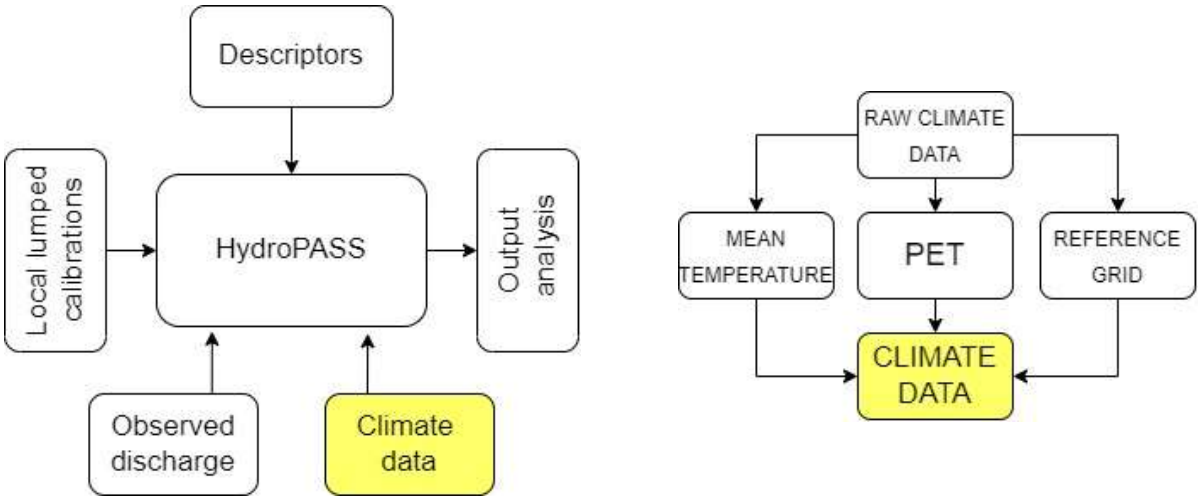


Figure 4 - R Studio environment subdivision

The use of this supporting interface was not mandatory, but indeed very useful. Nevertheless, the use of *R* from command line has been used at two different points: first, for the local lumped calibrations, then for the application of *HydroPASS*. In these two cases in fact the computational power (and so the required time) needed for the completion of the operations was so high that the use of my own personal device was insufficient. A remote access to better machines (that mounted not Window, but *LINUX* as operative system) was needed to complete the calculations in a reasonable timeframe.

From now on, no full description of the *R* codes used for the different steps described below will be provided to not dwell too much the length of this thesis (with a single exception later in chapter 6). Only the main functions and external packages used will be underlined whenever needed. After the conclusions in chapter 8 just the three main scripts for the core steps of this work, fully commented, will be appended to be copy pasted as template if anyone will ever need to reproduce them.

3. CLIMATOLOGICAL DATA AND DISCHARGE OBSERVATIONS



3.1 OI (optimum Interpolation) database and reference grid

The first element required to proceed both with the local lumped calibrations (chapter 4) and the application of the *HydroPASS* algorithm (chapter 6) is the definition of some input climate variables over the region of interest. For this specific work such variables are the mean temperature (T_m), the potential evapotranspiration (PET), and the precipitation (P). Note how additional input data (especially snow cover in the alpine region), may be also included. The data needed have been obtained from the *OI* (*Optimal Interpolation*) database of *ARPA Piemonte (http://www.idrologia.polito.it/web2/open-data/cd_Dati_Regione_Piemonte/50_anni_dati_meteo_Piemonte/index.htm).*

The database is mainly based on the *"optimal interpolation"* technique to obtain, starting from scattered (spatially speaking) and heterogeneous (different measuring instruments with different time resolutions) information, a complete and homogeneous dataset over the nodes of a regular grid. The main idea is that only the closest measurements to each node can significantly influence its final value in the grid. The values over the grid x_a are obtained with a linear relationship between the background value at each node x_b and the difference between the observed and the background values at each measurement point $(y_0 - y_b)$ properly weighted through a matrix K:

$$x_a = x_b + K(y_0 - y_b) \qquad (eq.3.1)$$

$$K = G(S + O)^{-1} \qquad (eq.3.2)$$

Where G is the covariance matrix between the error of the background field on the nodes and the background field over the measuring station's locations, S is the covariance error matrix for all the possible couples of measuring station's locations, and O is covariance error matrix of the observation. As background data, depending on the climatological data, different sources have been used: for temperatures a downscaling of the ERA-40 elaborated by the *ECMWF (European Centre for Medium range Weather Forecast)* has been utilized for the period 1957-2001, while a different set again from the *ECMWF* has been used for the following years. For precipitation it has not been possible to rely on a model as background, and so it has been calculated starting from the same observation with a detrending procedure.

The resolution (spatial and temporal) of the resulting homogeneous database has been selected starting from considerations about the available observation locations and data availability. Two data sources have been introduced in the optimal interpolation:

a) A manual and mechanical monitoring network *SIMN*
b) A telemetry network managed by Arpa Piemonte

The resulting optimal spatial resolution is of 0.125° (approximately a pixel on the ground of $12x12\ km$) and for the following elaboration a daily time resolution has been used. The grid consists of 500 pixels over the regional area of Piedmont. The original reference system of the raster data, as downloadable from the *ARPA* Piemonte website, was *WGS84* (a geodetic reference system), while the current project uses a cartographic projection in *WGS84-UTM32*. The use of a different reference system opens to a delicate problematic: the original raster files should be also reprojected to the new reference system, but in doing so a proper interpolation method should be selected. Common strategies are *"closest neighbor"* or *"bilinear interpolation"*. The approximations introduced during this procedure are the bigger the coarser is the resolution of the raster file. Having in our case a very coarse grid, this approach has been avoided. What has been done instead is a reprojection of the grid with a direct association of the original values of the raster files without any interpolation. In simple words the values of, for example, pixel-1 in *WGS84* have been maintained unmuted and passed to pixel-1 in *UTM32* and so on for all cells.

The reprojection of the grid from the first to the second reference system has been performed directly in the *R* environment with the package *"raster"* that introduce useful functions to manipulate reference systems and reproject raster and vector objects (in particular the function *"SpTransform"*):

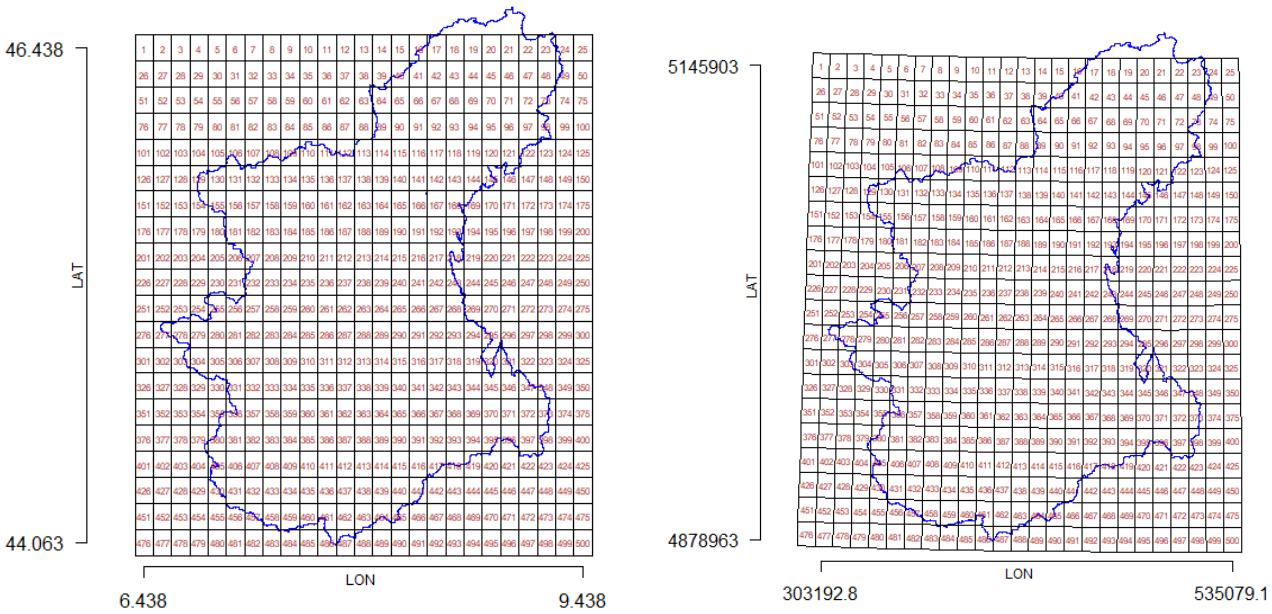


Figure 5 - Grids in WGS84 (left) and UTM32 (right)

It can be noticed how the upper part of Canton-Ticino is not covered by the adopted grid. As examples, the spatial distribution of the climate variables on the 07-01-1997 and the timeseries for pixel-77 are reported in Figures 6 and 7:

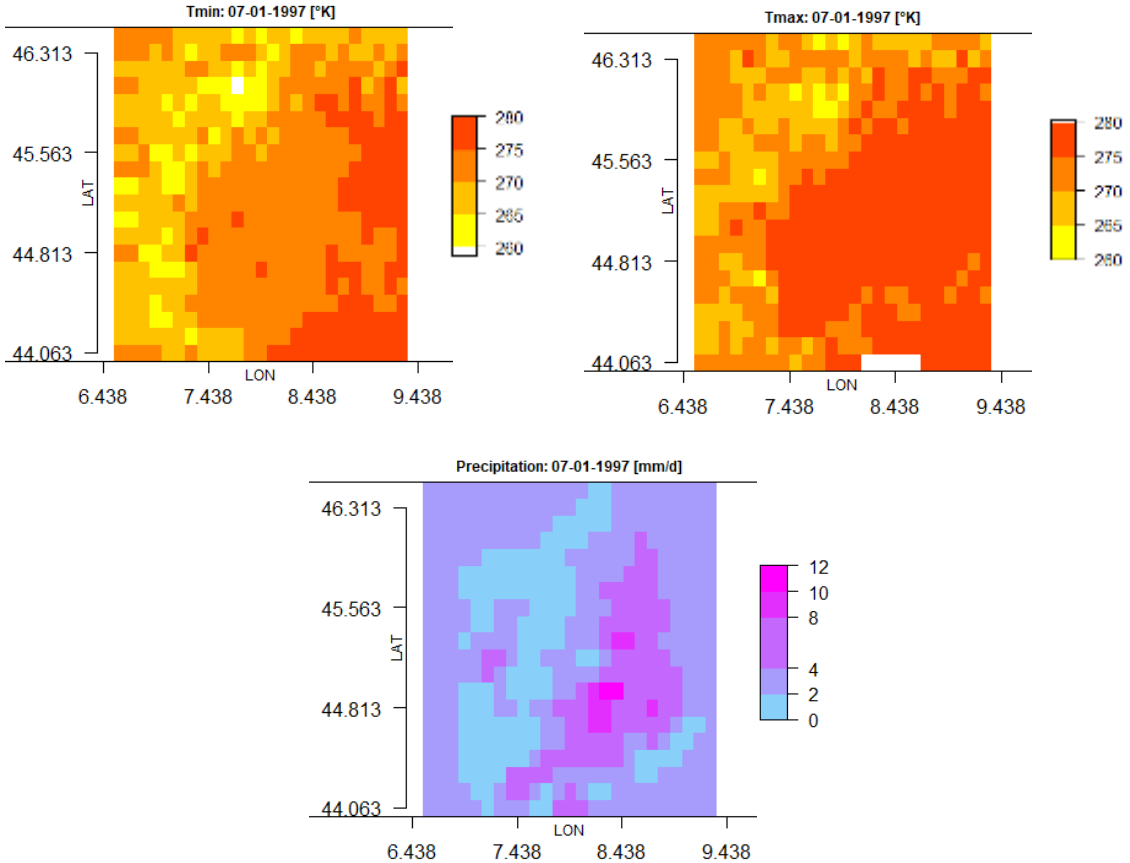


Figure 6 - Spatial distribution of Precipitation, Maximum temperature, and Minimum temperature on 7th January 1997

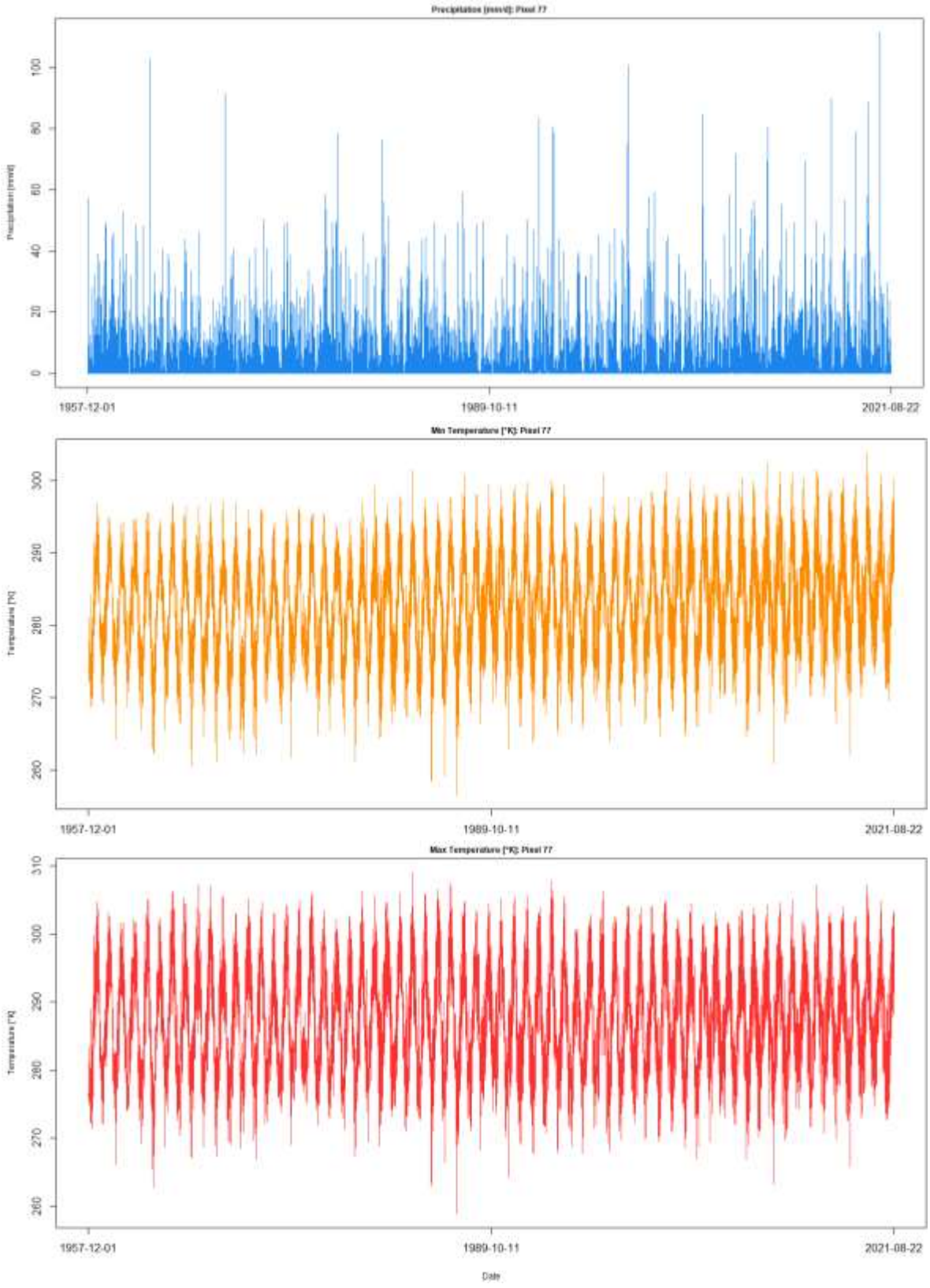


Figure 7 - Full precipitation, maximum temperature, and minimum temperature timeseries for pixel 77

The available timeseries extended across more than sixty years, starting from the 1st of December 1957 to the 22nd of August 2021.

3.2 Mean temperature and potential evapotranspiration

From the *OI* database the available data were the maximum daily temperatures ($T\_\max$), the minimum daily temperatures ($T\_\min$) and the precipitations (P). If the latter is already one of the three climate variables needed as stated at the beginning of the chapter, the other two must still be elaborated to obtain what is needed: the daily mean temperature T_{mean} can be easily obtained just by averaging the daily minimum and maximum temperatures. A different procedure must instead be followed for the potential evapotranspiration PET: no direct measurements have been performed, and the necessary data have been obtained as result of an indirect procedure. Different approaches and empirical equations are listed in scientific literature to infer the ET starting from other climatological data:

a) Radiation-based approaches require the mean average temperature and the solar incoming radiation; they are considered good in humid environments but not so reliable in arid ones.
b) The Penman-Monteith equation is the reference methodology applied by many international agencies (i.e., *FAO* -> Food and Agriculture Organization) and provides the optimal and most reliable results:

$$ET_0 = \frac{\Delta(R_n - G) + \rho_a c_p (\delta e) g_a}{\big(\Delta + \gamma(1 + g_a/g_s)\big) L_v} \qquad (eq.\,3.3)$$

   The major downside of this methodology is the huge amount of data not easily gettable required for the computation like the ground heat flux G or the conductivity of stomata g_s

c) The Blaney-Criddle equation is the preferred approach when not all the data for the correct application of the Penman-Monteith equation are available. More specifically it only requires the mean air temperature for a correct implementation. Given its coarse accuracy it is suggested for long term application (at least on a monthly timescale)

Considering our conditions, the preferred choice has been the application of the Blaney-Criddle equation following the indications and guidelines provided by FAO:

$$ET_0 = p(0.457 * T_{mean} + 8.128) \qquad (eq. 3.4)$$

Where ET_0 represents the *reference crop evapotranspiration*, p the *mean daily percentage of annual daytime hours* and T_{mean} the *mean temperature.* For the determination of the monthly p values inside the same guidelines the following table is provided for a correct evaluation:

Latitude	North	Jan	Feb	Mar	Apr	May	June	July	Aug	Sept	Oct	Nov	Dec
	South	July	Aug	Sept	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	June
60°		.15	.20	.26	.32	.38	.41	.40	.34	.28	.22	.17	.13
55		.17	.21	.26	.32	.36	.39	.38	.33	.28	.23	.18	.16
50		.19	.23	.27	.31	.34	.36	.35	.32	.28	.24	.20	.18
45		.20	.23	.27	.30	.34	.35	.34	.32	.28	.24	.21	.20
40		.22	.24	.27	.30	.32	.34	.33	.31	.28	.25	.22	.21
35		.23	.25	.27	.29	.31	.32	.32	.30	.28	.25	.23	.22
30		.24	.25	.27	.29	.31	.32	.31	.30	.28	.26	.24	.23
25		.24	.26	.27	.29	.30	.31	.31	.29	.28	.26	.25	.24
20		.25	.26	.27	.28	.29	.30	.30	.29	.28	.26	.25	.25
15		.26	.26	.27	.28	.29	.29	.29	.28	.28	.27	.26	.25
10		.26	.27	.27	.28	.28	.29	.29	.28	.28	.27	.26	.26
5		.27	.27	.27	.28	.28	.28	.28	.28	.28	.27	.27	.27
0		.27	.27	.27	.27	.27	.27	.27	.27	.27	.27	.27	.27

Table 1 - Mean daily percentage of annual daytime hours for different latitudes

For a latitude of $\sim 45°$ north (somehow an average of the Piedmont regional territory) p ranges from a minimum of 0.2 in the months of January and December to a maximum of 0.35 in the month of June. It must be underlined how this equation, particularly in "extreme" environments, has the tendency to overestimate or underestimate (in some cases considerably, up to $20/30\%$) the real evapotranspiration. Now because Italy is not considered a region exposed to "extreme" climate, we can reasonably assume this is not our case, but considerations and questions should eventually be posed while analyzing the results. Like for the original climate data, in Figure 8 is reported the timeseries of the PET for pixel-77:

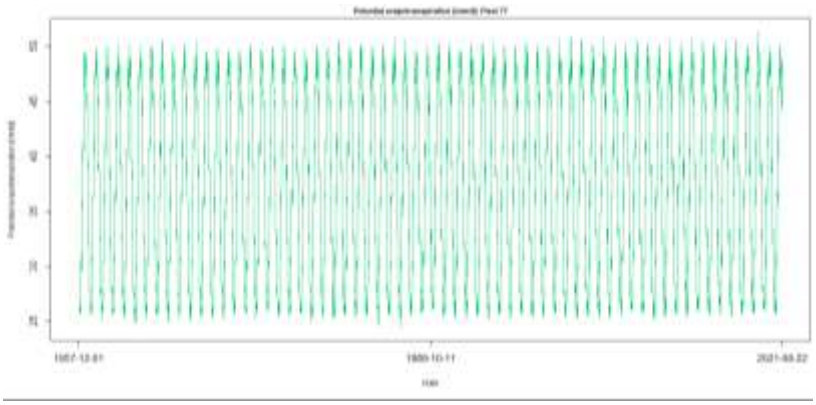
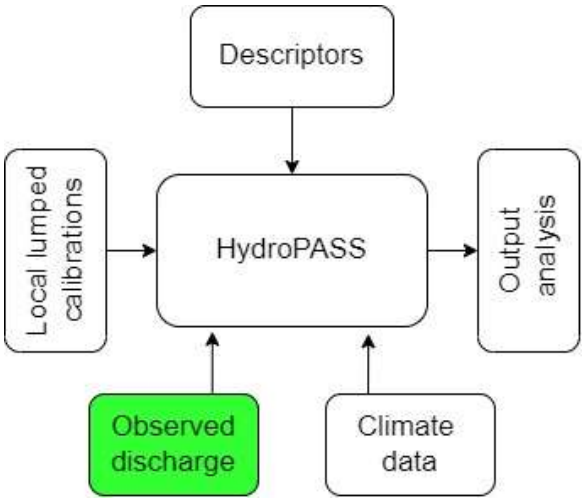


Figure 8 - Full timeseries of the potential evapotranspiration (PET) for Pixel-77

3.3 Observed discharge and catchments description



Together with the climatological data, the second observation variable needed for the *HydroPASS* algorithm are the observed discharges for the gauged catchments considered over the regional territory. The shape files of the catchments analyzed inside this work have been recovered from *"Atlante dei bacini imbriferi piemontesi" (Gallo, E., Ganora, D., Laio, F., Masoero, A., Claps, P., 2013)* and consist of 197 boundaries in vector format already projected in the *UTM32* reference system (and so coherent with the reference system of the entire project) as reported in Figure 9:

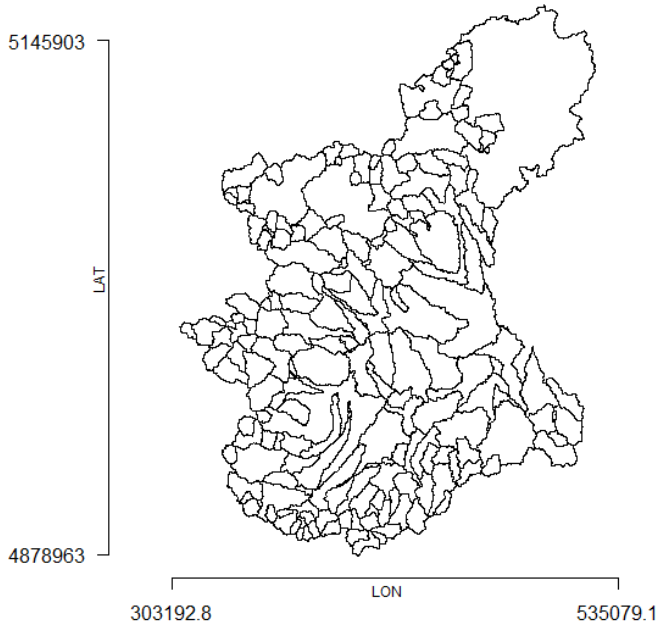
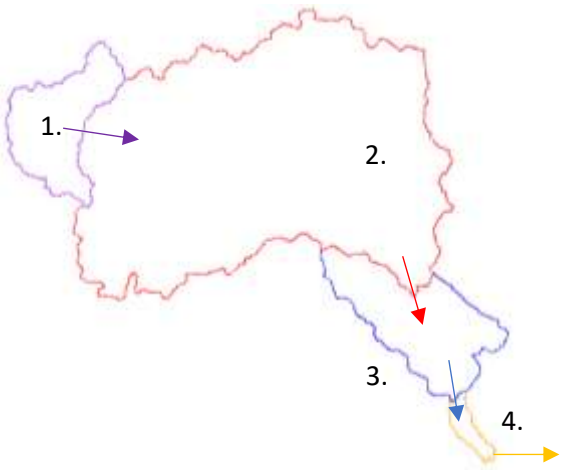


Figure 9 - Catchment boundaries Piemonte and Valle d'Aosta regions

A catchment (also called drainage basin) is defined as any area on land where precipitation is collected and drains towards a single common outlet. Following this definition, a catchment does not present, in general, any constrains about the number of inlets it may present. As will be better explained in chapter 4, inside the hydrological model selected for the local lumped calibrations the inter-catchment water routing is not considered (it is only the intra-basin routing), disregarding the movement and timing of flow from an upstream to a downstream basin. To be coherent with the model, our catchments, still obeying to the standard definition of having a single outlet, present an additional constrain of never receiving water from any basin upstream.

To make this concept clear is here reported a simple example case of an allowed configuration:



Basin 1 does not receive water from upstream; basin 2 (maybe not clear from the image but delimited by the red ***AND*** the purple boundaries) also has no inlets and so on (catchment 3 is delimited by the blue + red + purple boundaries and catchment 4 by the orange + blue + red + purple ones). A configuration like this, constituted by ***"concentrical"*** areas, satisfies both the single outlet and the zero-inlet condition. In mathematical terms, in case of a simple linear succession of basins like the one in the explanatory case above, the following relation must be satisfied:

$$C_m \supset C_i \quad \forall\, i = 1{:}\,n \qquad\qquad (eq.\,3.5)$$

$$with\ 1\ the\ most\ downstream\ and\ n\ the\ most\ upstream\ catchments$$

Information about each outlet location and additional data (*ID* codes, names, area, and some statistical information) have been again retrieved from the *"Atlante dei bacini imbriferi Piemontesi" (Gallo et al., 2013).* The information about the area (expressed in squared kilometers), has been of fundamental importance to convert the discharge data from m^3/s to mm/d. This conversion has been necessary to make the runoff dimensionally coherent with precipitation and evapotranspiration data, already expressed in mm/d. The discharge timeseries have been obtained merging the available data downloadable from the *ARPA Piemonte* website with additional data covering the Valle d'Aosta region.

These timeseries are very heterogeneous, in the sense that some are limited to few years while other are comprehensive of historical data, some are mostly complete, others present huge gaps due to interruptions or malfunctioning of the monitoring systems. These differences will play a role once again in the local lumped calibration as explained in further details in chapter 4. Moreover, discharge observations were available only for 127 of the 197 catchments identified before. In Figures 10 and 11 are reported a map of the catchments with the location of all the outlets and a couple of examples of raw discharge timeseries:

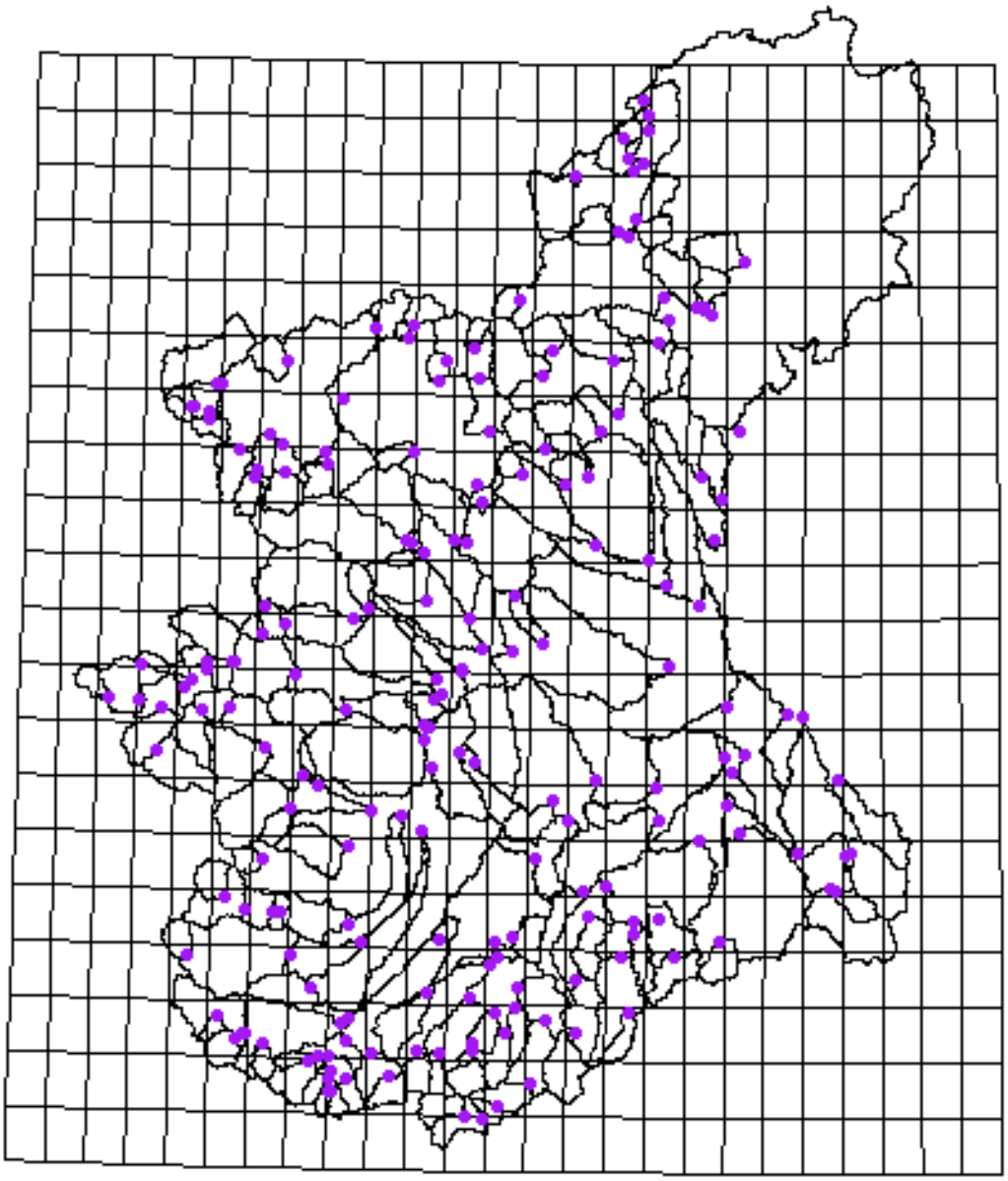


Figure 10 - Map of the catchment's outlets in UTM32 projection with reference grid

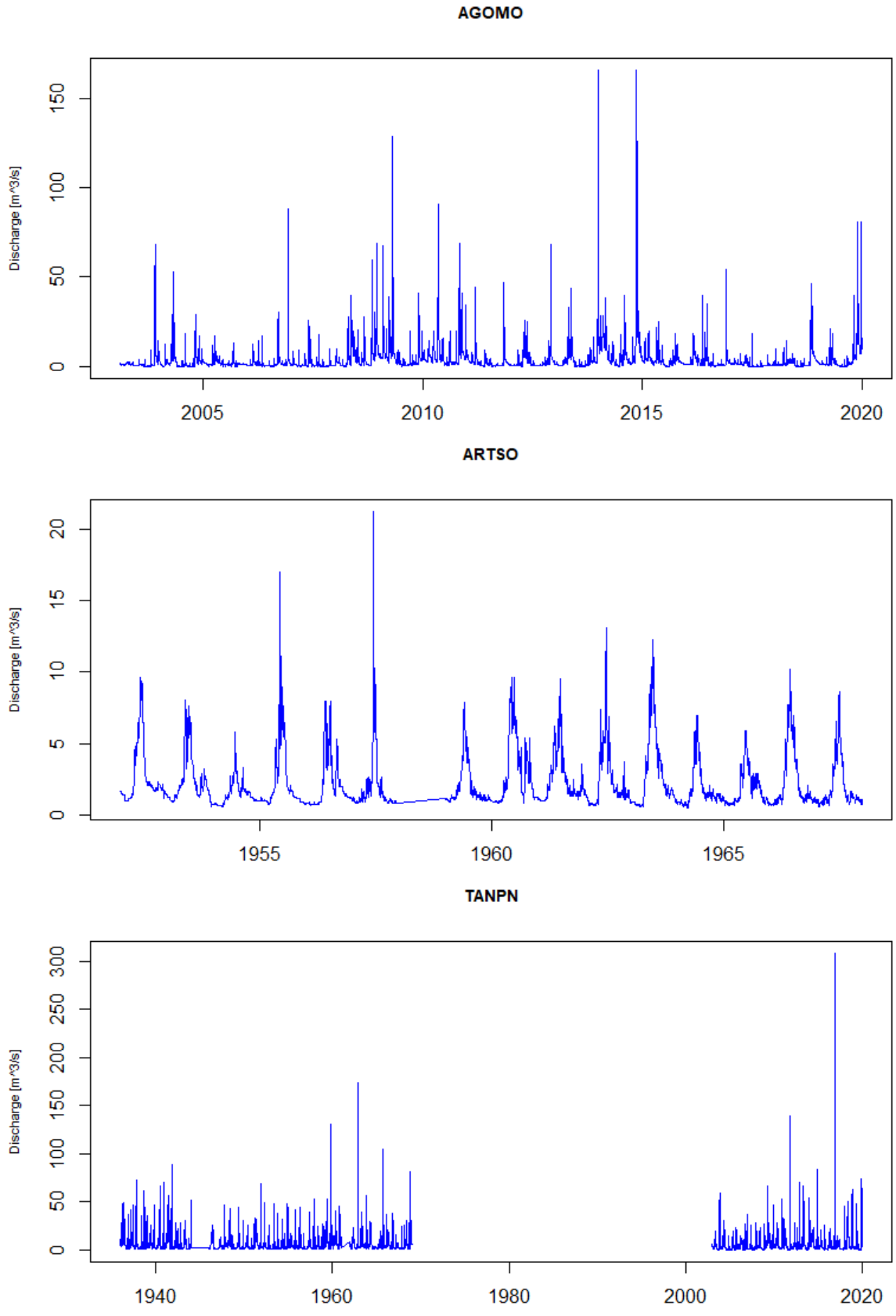
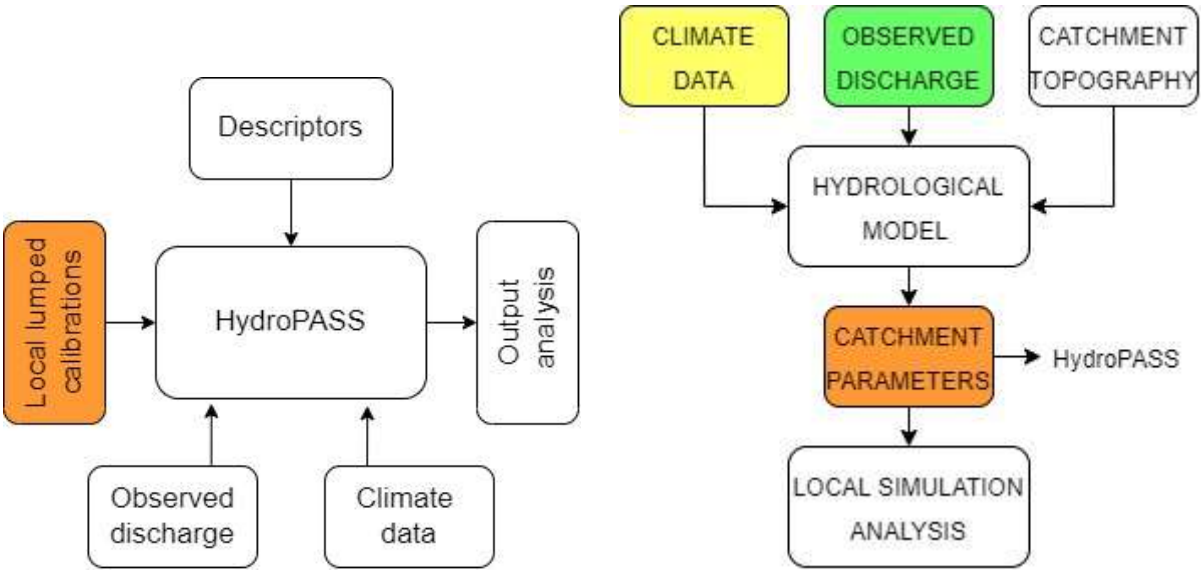


Figure 11 - Examples of a complete discharge timeseries (above), historical discharge timeseries (middle), and incomplete discharge timeseries (bottom)

4. LOCAL CALIBRATIONS



4.1 Hydrological models classification

In literature various models are proposed to face and "tackle" hydrological problems, in particular the rain-runoff correlation and prediction that constitutes one of the hardest challenges in environmental prevision and forecast. *"This is due to the spatial and temporal variability of topographical characteristics, rainfall patterns, and the number of parameters to be derived during the calibration"* (Nandakumar & Mein, 1997).

"Hydrologic models can be classified into categories based on the presence of random variables, their distribution in space, and temporal variation" (Chow, Maidment, & Mays, 1988):

1. ***DETERMINISTIC MODELS***: they are defined as models that produce always the same outputs given the same inputs. They can be further subdivided as:

    a) ***LUMPED MODELS***: *"a lumped model is generally applied to a single point or a region without dimension for the simulation of various hydrological processes"* (Niel, Paturel, & Servat, 2003). The parameters represent a sort of average of the characteristics of the entire catchment considered. Among the major advantages there is their simplicity and as consequence the lighter computational power (and so time) required for the simulations.

b) ***SEMI-DISTRIBUTED MODELS***: the catchment is here subdivided in Hydrologic Response Units (*HRUs*) based on considerations related to land cover, soil types and other parameters. The simulation is performed on each unit individually.

c) ***DISTRIBUTED MODELS:*** *"It considers the hydrological processes taking place at each grid and defines the model variables as functions of the space dimensions"*(Beven, Warren, & Zaoui, 1980; Feyen, Vázquez, Christiaens, Sels, & Feyen, 2000).

2. ***STOCHASTIC MODELS***: they are models with embedded a component of randomness due to which for the same input data multiple outputs can be obtained.

"Hydrological models can be also classified according to whether the hydrological processes are described as conceptual, empirical, or fully physically based" (G.S. Dwarakish,B.P. Ganasri, 2015). In the case of empirical models, no physical transformation function is directly introduced, but rather the relations between precipitation and runoff are established starting from the data themselves. Conceptual models are a simplification of the real physical processes that rule the rainfall-runoff transformation through a set of equations: such equations are generally derived by empirical observations of the real phenomena happening over a catchment. Last the physically based models are constructed upon differential equations that aim to mimic as rigorously as possible the physical phenomena controlling the runoff generation. A schematic representation of the described classification is reported in Figure 12:

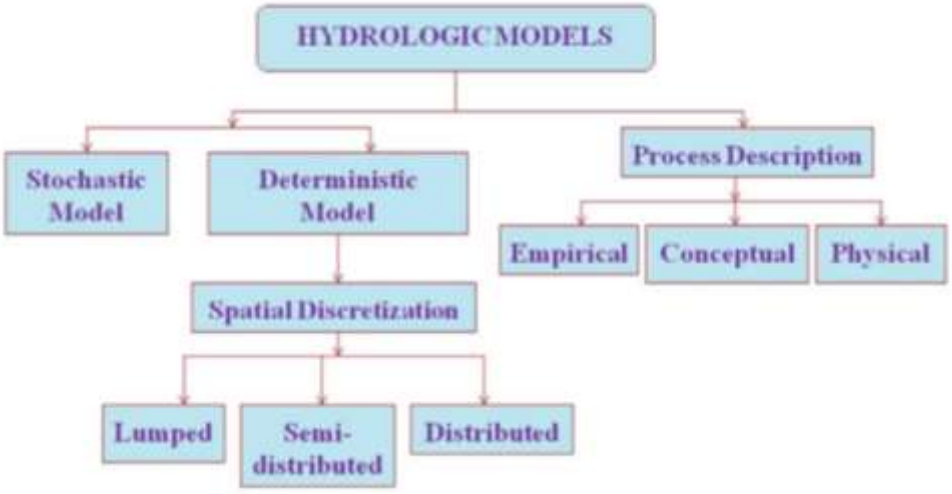


Figure 12 - Schematic representation of the hydrological models classification (G.S. Dwarakish,B.P. Ganasri, 2015)

4.2 TUWmodel

Inside the presented framework, a choice of one hydrological model was necessary as one of the main elements that would have influenced the outcome of the entire procedure. In the original article by Merz et al. of 2020 (already cited in the introduction), the selection fell on the ***SALTO*** (Same Like The Others) model, a deterministic distributed conceptual model with 15 parameters that uses daily timeseries of precipitation, potential evapotranspiration and air temperature as inputs. Despite in many parts of this work we have tried to remain as close as possible to the original study, the use of *SALTO* resulted, after some initial attempts, in weak calibration outcomes. Being the quality of the local catchments calibration like a starting point for the *HydroPASS* algorithm, after which the overall efficiency will inevitably reduce, it was unacceptable the idea of starting already from bad results. The reason behind the poor performance of this model is to be probably attributed to the region we are trying to apply the methodology to: the model was originally developed to investigate German catchments, in a non-mountainous area where the influence of snow and solid precipitation have a lower influence on the hydrological cycle. In an Alpine region like the one here considered, a different model developed and structured to better account for this major difference was more suitable. Following this consideration, the model selected has been ***TUWmodel***, a lumped conceptual rainfall-runoff model (that can also run in a semi-distributed mode) developed at the Vienna university of technology following the structure of the ***HBV*** model. The model runs on a daily (like in our case) or shorted timeframe for the input and output data and consists of a snow routine, a soil moisture routine, and a flow routing routine. The description of the model reported below follows the *Appendix A* of *"Uncertainty and multiple objective calibration in regional water balance modelling: case study in 320 Austrian catchments*" (*J. Parajka, R. Merz and G. Bloschl, 2007*).

A) <u>SNOW ROUTINE</u>

This first component of the model is based on a simple *"degree-day"* snow melt and accumulation concept. The daily precipitation P is partitioned in liquid (rain) and solid (snow) depending on the mean air temperature T_A at the current timestep of the simulation and the relative "position" with respect to two reference temperatures T_R (rain temperature) and T_S (snow temperature):

$$P_R = P \qquad if\ T_A > T_R \qquad (eq\ 4.1a)$$

$$P_R = P * \frac{T_A - T_S}{T_R - T_S} \quad if\ T_S < T_A < T_R \quad (eq.4.1b)$$

$$P_R = 0 \quad if\ T_A < T_S \qquad (eq.4.1c)$$

$$P_S = P - P_R \qquad (eq.4.1d)$$

The melting process instead starts when the mean air temperature T_A overcomes the melting temperature threshold T_M:

$$M = (T_A - T_M) * DDF \quad if\ T_A > T_M\ and\ SWE > 0 \qquad (eq.4.2)$$

Where M is the amount of melt water per time step, DDF is a *degree-day factor* and SWE is the *snow water equivalent.* Of course, melting can only happen when there is water stored in solid form. Moreover, the SWE at each timestep is corrected through an additional parameter (*snow correction factor* SCF) accounting for the catch deficit of the precipitation gauges during snow fall following the relation:

$$SWE_i = SWE_{i-1} + (SCF * P_S - M) * \Delta t \qquad (eq.4.3)$$

With Δt a timestep of one day. In total five parameters are used to describe this first routine $(T_R, T_S, T_M, DDF, SCF)$.

B) <u>SOIL MOISTURE ROUTINE</u>

This second routine describes changes in the moisture content of the basin and runoff generation. The first of the two is describes as:

$$S_{SM,i} = S_{SM,1-i} + P_R + M - E_A \qquad (eq.4.4)$$

S_{SM} represents the soil moisture at each time step at a top-soil layer as the balance between the soil moisture at the previous timestep, the additional rain plus melt water, minus the actual evapotranspiration. It is the top-soil layer to contribute to runoff generation. The actual evaporation E_A is determined starting from the potential evapotranspiration PET following the following linear function:

$$E_A = PET * \frac{S_{SM}}{LP} \quad if\ S_{SM} < LP \qquad (eq.4.5a)$$
$$E_A = PET \quad if\ S_{SM} \geq LP \qquad (eq.4.5b)$$

LP is a parameter controlling the limit for potential evaporation PET.

The direct contribution of rain and water melt to runoff is expressed through a non-linear relation controlled by two additional parameters:

$$\Delta S_{UZ} = \left(\frac{S_{SM}}{FC}\right)^{\beta} (P_R + M) \qquad (eq.4.6)$$

Where ΔS_{UZ} is the contribution of rain and snow melt to runoff, FC is the maximum soil moisture storage and β is a non-linear parameter that controls the characteristics of runoff generation. When the ration between S_{SM} and FC is equal to 1 (the soil il fully saturated), all the melt water and precipitation contributes directly to runoff generation. In total we have three additional parameters.

C) ***RESPONSE AND TRANSFER FUNCTION ROUTINE***

The last component of the model deals with runoff routing inside the catchment on the hillslopes and consists of two reservoirs representing two soil layers. The state of each reservoir, at each timestep (i.e., the water stored in them), are S_{UZ} for the upper zone and S_{LZ} for the lower zone instead. The water balance of the upper reservoir is influenced by the direct inflow due to precipitation and water melt already described ΔS_{UZ} and by three outflows: the first with a fast *storage coefficient* K_1, the second representing soil percolation to the lower zone controlled by a constant *percolation rate* C_p, and the third (active only when the storage in the upper zone overcome a specific threshold $lsuz$) controlled by a different *storage coefficient* K_0. The balance of the lower zone is instead determined by the input percolation from the upper layer plus an outflow controlled by a *slow storage coefficient* K_2. The sum of the three outflows controlled by the storage coefficients determines the total runoff Q_G. This is then routed by a triangular transfer function representing the runoff routing in the streams:

$$B_Q = B_{MAX} - C_R Q_G \quad if\ (B_{MAX} - C_R Q_G) \geq 1 \qquad (eq.4.7)$$
$$otherwise\ B_Q = 1$$

Where B_Q is the base of transfer function, B_{MAX} is the maximum base at low flows and C_R is a free scaling parameter. With the seven parameters of this last routine the total parameters to be calibrated for the model rises to fifteen.

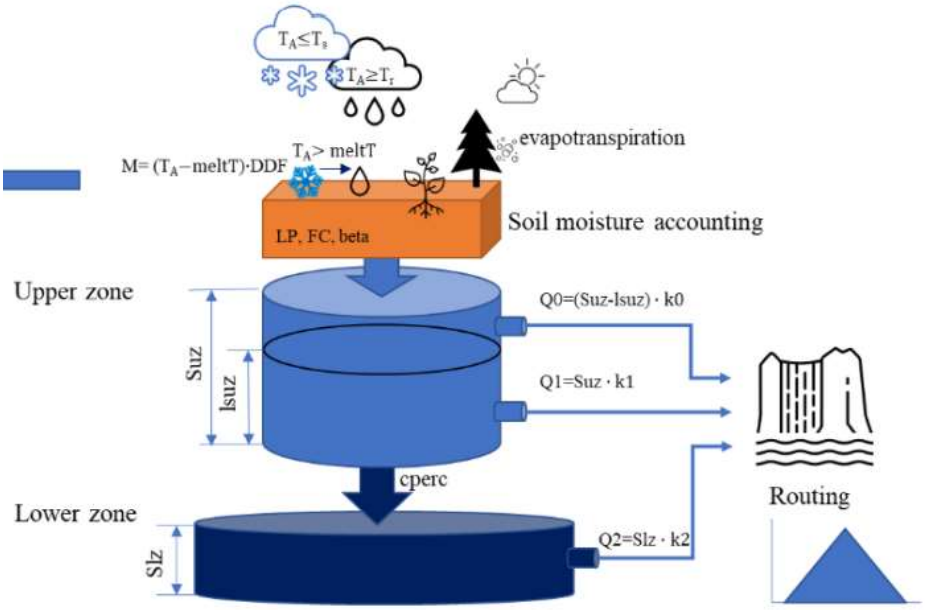


Figure 13 - Conceptual description of TUWmodel structure (Rui Tong, Juraj Parajka, 2020)

4.3 Local lumped calibrations – DEoptim

The main aim of this section is the determination, for each of the gauged catchments presented in the paragraph 3.3, of the best set of the 15 lumped parameters that can better reconstruct the observed discharge timeseries at the outlets. In optimization processes with best is intended the set of parameters that maximize (or minimize in some cases) an objective function appositely constructed to verify the performance of the model. To perform such operation different methods can be applied as well as different model efficiency function can be utilized inside such methods. For the current work a Differential Evolution (*DE*) algorithm, as described by Muller et. al in 2011, has been utilized: it is an evolutionary global optimization implemented inside the *R* environment through the specific package

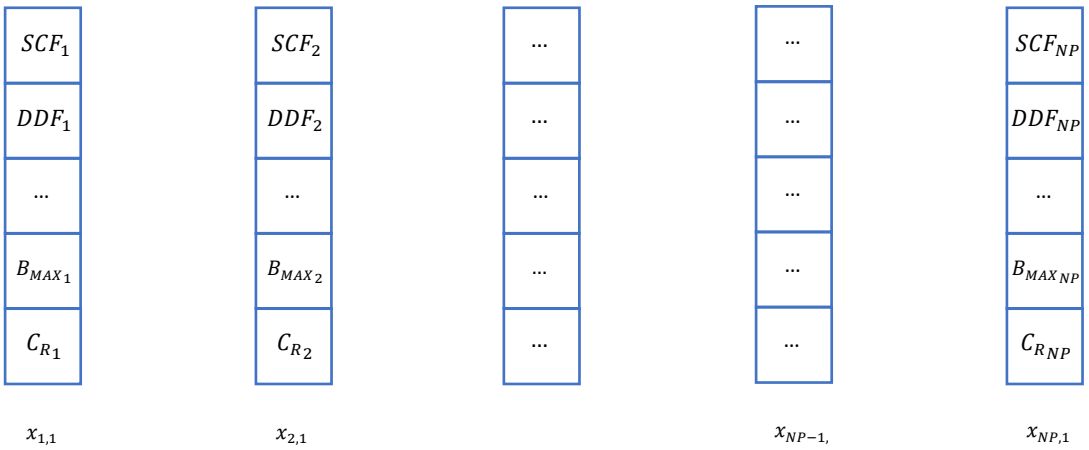
"DEoptim" first published by David Aria back in 2021. Differential evolution falls within the genetic optimization algorithms "*in which members of the population are represented with floating point numbers, and the population is transformed over successive generations using arithmetic operations"* (Mullen, 2011). One of the main advantages of *DE* is the fact that it does not require the differentiation of the objective function and can so be successfully applied to all the cases when the latter is noisy, stochastic, non-continuous or non-differentiable. On the other hand, DE may result inefficient whenever the objective function is indeed differentiable and "smooth"; in those cases, a standard derivative-bases approach would reach better results.

Now follows a short description of the evolutionary algorithm starting from the determination of the space of the possible solutions and the size of the population. The first is determined by an upper and lower boundary for each of the 15 parameters of *TUWmodel* as reported in Table 2:

PARAMETER	LOWER BOUNDARY	UPPER BOUNDARY
$\boldsymbol{SCF}$ $[-]$	0.9	1.5
$\boldsymbol{DDF}$ $[\boldsymbol{mm/°C/day}]$	0	5
$\boldsymbol{T_R}$ $[\boldsymbol{°C}]$	1	3
$\boldsymbol{T_S}$ $[\boldsymbol{°C}]$	-3	1
$\boldsymbol{T_M}$ $[\boldsymbol{°C}]$	-2	2
$\boldsymbol{LP}$ $[-]$	0	1
$\boldsymbol{FC}$ $[\boldsymbol{mm}]$	0	600
$\boldsymbol{\beta}$ $[-]$	0	20
$\boldsymbol{K_0}$ $[\boldsymbol{days}]$	0	2
$\boldsymbol{K_1}$ $[\boldsymbol{days}]$	2	30
$\boldsymbol{K_2}$ $[\boldsymbol{days}]$	30	250
$\boldsymbol{lsuz}$ $[\boldsymbol{mm}]$	1	100
$\boldsymbol{C_p}$ $[\boldsymbol{mm/day}]$	0	8
$\boldsymbol{B_{MAX}}$ $[\boldsymbol{days}]$	0	30
$\boldsymbol{C_R}$ $[\boldsymbol{days^2/mm}]$	0	50

Table 2 - Upper and lower boundaries for TUWmodel parameters

The second (NP) can be selected by the user. Once selected the size of the population, at the first generation a number of x- dimensional vector (where x is the number of parameters) equal to the number of individuals is randomly generated inside the space of the possible solutions:



To pass from a generation to the successive, each vector of the previous generation is *"differentially mutated".* The new potential *"mutant"* parameters vector is created by choosing at random three vectors of the current population ($x_{r0,g}$, $x_{r1,g}$, $x_{r2,g}$ with g indicating the current generation) applying the following equation (here as reference let us assume to mutate the first individual of the first generation):

$$v_{1,1} = x_{r0,g} + F\left(x_{r1,g} -, x_{r2,g}\right) \qquad (eq.\,4.8)$$

Where F is a positive scaling factor usually lower than 1. After the first mutation, the resulting mutant parameters vector can be further modified following the same approach until:

$$rand > CR$$

Where $rand$ is a random number extracted from a uniform distribution $U(0,1)$ and CR is a crossover probability $\in [0,1]$. The final mutant vector may still be unacceptable because the equation 4.8 does not guarantee that the resulting parameters are still inside the upper and lower boundaries defining

the valid solutions space. In these cases, the not admissible elements of the vector are reset in the following way:

$$v_j = upper_j - rand * \left(upper_j - lower_j\right) \qquad if\ v_j > upper_j \qquad (eq.4.9a)$$

$$v_j = lower_j + rand * \left(upper_j - lower_j\right) \qquad if\ v_j < lower_j \qquad (eq.4.9b)$$

Where the index j denotes the position inside the mutant vector of the element overcoming the boundaries. Finally, the new vector is used to evaluate the objective function: if the computed value is lower than the one of the original parameters vector from the previous generation, the new one substitutes it. The same procedure is applied to all the NP vectors of the generation and for all the generations until the maximum number of iterations decided by the user have been reached (in this case the best solution is the one of the members of the current population that minimize the most the objective function), or when a threshold value of model efficiency has been achieved by any individual of any generation. There is still a third possibility in case the algorithm is unable to improve the solution of the previous generation of a selected threshold: here the optimization algorithm stops, and as best solution is considered the better found till that point.

A final note is about the reproducibility of the results: being the starting guess at the first iteration performed in a random way, starting from the same input data, different outputs may be reached. It is also possible that for a given calibration, what is reached is not the global minimum of the objective function, but a local one. The only way to increase the probability of not remaining stuck in a local solution, is to perform more than one calibration attempt hoping to reach new and better calibrations.

4.4 Local lumped calibrations – application and objective function

Moving to our specific case, the lumped calibration of the 127 catchments having observed discharge timeseries as described in chapter 3 has been performed following the approach:

a) Determination of the objective function for the evaluation at each generation of the ME (Model Efficiency). This function is the same for all catchments.
b) Extraction, for the current catchment under calibration of what, from now on, will be referred to as *"train topology"*. Different weights (from 0 to 1) are attributed to different pixels. They quantify the proportion of the pixels that cover the catchment of interest.

c) Extraction from the climate input data (T_{MEAN}, P and PET) only of the pixels covered by the catchment considered (only the pixels with a weight greater than 0).

d) Application of the DE algorithm with the "*DEoptim"* package inside *R* for the determination of the model parameters. Being the calibration lumped the sets found for each basin are representative of the average conditions over the entire covered area. For each generation the sets of parameters are tested with the direct application of *TUWmodel* to determine a simulated discharge timeseries used inside the objective function with the observed discharge data to obtain a model Efficiency (ME). It is an iterative process to refine the solution proceeding with successive generation till the determination of the optimal solution.

"Model performance criteria are often used during calibration and evaluation of hydrological models to express in a single number the similarity between observed and simulated discharge" (Gupta et al., 2009). The model efficiency here utilized is the Kling-Gupta efficiency, first developed in 2009 and then proposed in a revised version in 2012 by Gupta et al. It is based on the decomposition of an older metric, the Nash-Sutcliffe efficiency (NSE, Nash and Sutcliffe, 1970), into its constitutive components:

$$NSE = 1 - \frac{\sum_{t=1}^{t=T}[Q_{sim}(t) - Q_{obs}(t)]^2}{\sum_{t=1}^{t=T}[Q_{obs}(t) - \overline{Q_{obs}}]^2} \qquad (eq.\,4.10)$$

Where T is the total number of time steps, Q_{obs} the observed discharge, Q_{sim} the simulated discharge and $\overline{Q_{obs}}$ the mean of the observed discharge.

The KGE metric is composed by three distinct elements, representing different statistics of the two compared data series: the Pearson product-moment correlation coefficient (r) which optimal possible value equal to 1, the ratio between the mean of the simulated and observed data (β) again equal to 1 in case of a perfect match, and the so-called variability ratio (vr) that can be computed using the standard deviation (α) or the coefficient of variation (γ) of the two series:

$$KGE = 1 - ED$$

$$ED = \sqrt{(r-1)^2 + (\beta-1)^2 + (\gamma-1)^2} \quad or \quad ED = \sqrt{(r-1)^2 + (\beta-1)^2 + (\alpha-1)^2}$$

$$\beta = \frac{\mu_{SIM}}{\mu_{OBS}}$$

$$\gamma = \frac{CV_{SIM}}{CV_{OBS}} \quad or \quad \alpha = \frac{\sigma_{SIM}}{\sigma_{OBS}} \quad (eq.set\ 4.11)$$

The Kling-Gupta efficiency can range between -inf and 1, where unity indicates perfect agreement. To be clear, the actual model efficiency utilized is the negative efficiency (–KGE) because the DE algorithm is constructed to minimize, and not maximize, the objective function. The negative Kling-Gupta ranges between -1 and +inf. In literature values of the KGE are considered good (and so able to well simulate the observed discharge) if they are at least greater than 0, but preferably higher and closer to 1. Rogelis et al. (2016) considered model performance to be poor if comprised between 0 and 0.5. The assumption of zero as reference value to separate good and bad values derives from the original NSE where it corresponds to use the mean flow as benchmark predictor; nevertheless, other authors showed how for the Kling-Gupta using the mean flow as a predictor result instead in a value of $1 - \sqrt{2} \cong -0.41$. *"Thus, KGE values greater than* -0.41 *indicate that a model improves upon the mean flow benchmark even if the model's KGE value is negative"* (Knoben, W. J. M., Freer, J. E., and Woods, R. A.: Technical note: Inherent benchmark or not? Comparing Nash–Sutcliffe and Kling–Gupta efficiency scores, 2019).

Inside the *R* environment the *KGE* and its component have been calculated directly through the utilization of the additional package *"hydroGOF"* (version $0.4\text{-}0$, 2020) created and maintained by Mauricio Zambrano Bigiarini.

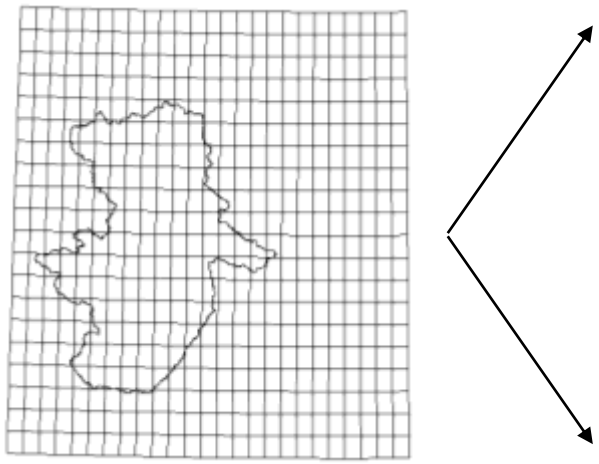
Regarding the *"train topology"* instead, the *"sf" R* package has been used. It is an additional package of functions that like *"raster"* introduces numerous ways of manipulating raster and vector elements. Differently from *"raster"* some more direct and intuitive functions for the manipulation of shape files are coded. First, the original vectors file of the catchment's boundaries (Figure 9) and of the grid (Figure 5) have been exported in *.shp* format (package "*rgdal"*) and reimported with the *"sf"* import function ("*st_read"*) inside the *R* environment. Then the intersection of the two has been calculated ("*st_intersection"*) to obtain the area (expressed in m^2) of each pixel touched by the catchments. To summarize the results the additional function *"summarize"* form the *"tydiverse"* package has been utilized.

As stated above, the desired *"train topology"* must be expressed in percentage terms. To do so is needed particular attention. The original grid in *WGS84* from the *OI* database was regular, with all the pixels having the same exact dimension. After the reprojection in *UTM32*, the new pixels are no regular anymore, being slightly affected by deformations; as direct consequence, the area of each single pixel is different from the one of the neighbors, as it decreases moving eastward and increases moving

southward. To account for this aspect, the area of each single pixel has been evaluated starting from the grid shape file (function "*st_area"*) and each weight then accordingly normalized:

$$W_{i,j}[\%] = W_{i,j}[m^2]/A_i[m^2] \quad (eq.4.12)$$

Where the index i identify the pixel and the index j identify the catchment. In Figure 14 is reported a visual example for the *POCM* catchment:



	codice	Pxl	area
	<chr>	<int>	<dbl>
1	POCM	105	390509.
2	POCM	106	25550520.
3	POCM	107	6290628.
4	POCM	108	36621889.
5	POCM	109	55641769.
6	POCM	110	104296887.
7	POCM	111	84456312.
8	POCM	112	49293541.
9	POCM	129	52516720.
10	POCM	130	113958129.

... with 128 more rows

	<chr>	<int>	<dbl>
1	POCM	105	0.00290
2	POCM	106	0.190
3	POCM	107	0.0467
4	POCM	108	0.272
5	POCM	109	0.413
6	POCM	110	0.775
7	POCM	111	0.627
8	POCM	112	0.366
9	POCM	129	0.389
10	POCM	130	0.844

... with 128 more rows

Figure 14 - Example of "Train topology" [m^2 and %] for the POCM catchment

Some final considerations must be done about the methodology and application of the local calibration: the first one is still related to the grid; as already visually clear from Figure 5, not all the 500 pixels composing the grid intersect at least one of the catchments of the area considered for the analysis. To reduce the dimension of the datasets used inside the model, the climatological data related to such pixels have been removed and a grid containing just 305 pixels (see Figure 15) have been maintained moving forward. No changes in terms of internal pixels numeration have been caused by this operation (just the first pixel of the new grid is pixel-16 and not pixel 1, and the final pixel is not pixel-500 but pixel-488):

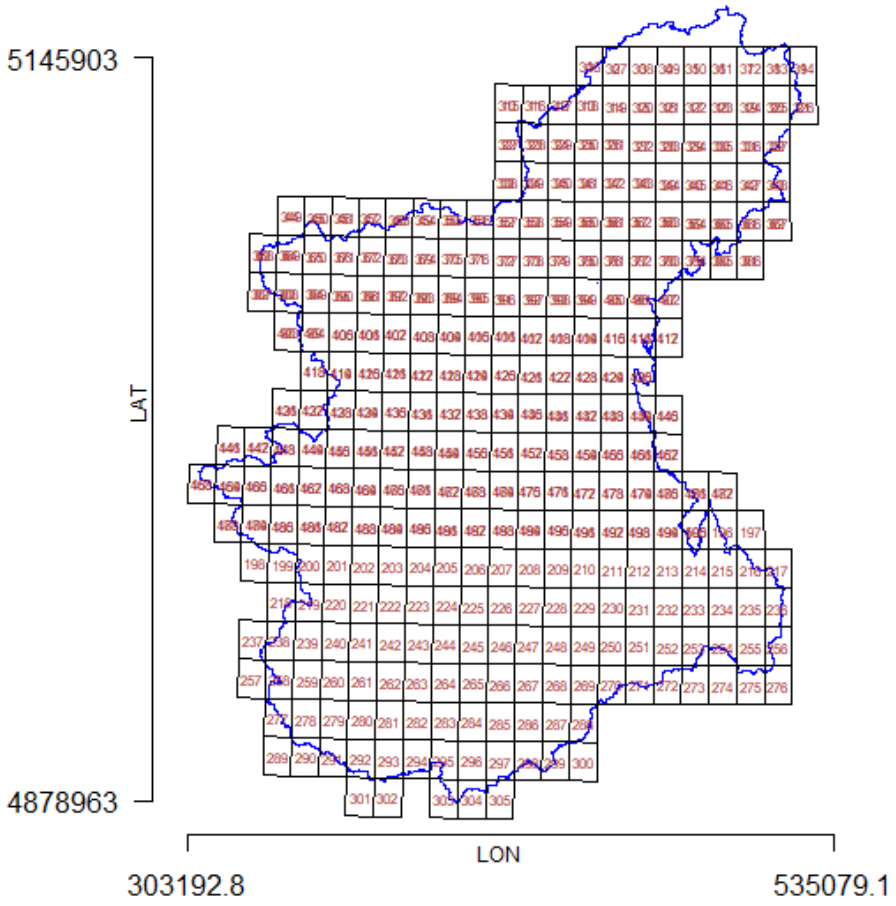


Figure 15 - Grid in UTM32 reduced to 305 pixels

The second comment is about the years considered for the analysis: despite the availability of data starting from 1957 arriving to 2021, it has been made the choice of limiting them to the period between the 1st of January 1961 and the 31st of December 2020 (exactly 60 years of data). The next comment is about the objective function: *TUWmodel* recreates the simulated discharge for all the 60 years; as described previously in the model structure, elements like the moisture content in the upper zone are influenced at each timestep by the value at the previous one. At the beginning of the simulation (first time step or time zero), no water was stored and the content at the second timestep was just the results of the balance of precipitation, water melt and evapotranspiration. This fist values, as well as all the first period of the simulation, are not realistic because biased by nonrealistic initial conditions. To avoid wrong, unmeaningful or partially biased evaluation of the ME, a first period of each simulated timeseries of 303 days (the warmup period) has been neglected from the computation. Also, all the eventually present days related to missing values (NA) inside the observed discharge have been neglected while computing the *ME*.

The next point is related to the simulated results used inside the objective function for the determination of the ME: *TUWmodel* is not only able to simulate discharge, but also other timeseries like soil moisture content and notably snow cover. In a mountainous region like the Alpine area snow accumulation could be a useful and meaningful parameter to be introduced while assessing the efficiency of the model reconstruction capability. Having available also snow cover observation timeseries could enable this additional comparison introducing modification in the objective function like $ME = 0.5 * KGE + 0.5 * ADD$, with *ADD* as an additional index related to the snow water equivalent as calculated by the model and the observed snow cover. Such choice is certainly possible, but we decide to concentrate on the simplest case without it.

Finally, are here briefly reported the settings utilized for the *DE* algorithm: a dimension of the population of 50 (default value), a maximum number of iterations (generations) of 200, a relative convergence tolerance of 10^{-3} applied after at least 10 generation has passed.

4.5 Local lumped calibrations – Results and not calibratable basins

The local calibration process has been in most cases successful, but not always possible. Some gauged catchments had an observed discharge timeseries only historical and prior to 1961. In those cases, the whole procedure has not been possible due to the missing of an overlapping time period longer than 303 days (the warmup period) with the simulated discharge series. Overall, the calibration has been possible for 117 basins. A rough classification of the goodness of the calibrations has been made arbitrarily using the following criteria:

$$Good\ if\ ME \geq 0.85$$
$$Moderate\ if\ 0.85 > ME \geq 0.75$$
$$Bad\ if\ ME < 0.75$$

As already stated during the initial presentation of the *KGE*, also a value of 0.6 (even below the lower threshold here considered) is far above the mean flow benchmark 0f $\cong$-0.41, but to hope in good results after the application of the regionalization routine, the highest the *MEs* of the calibration used as input the better. The output obtained after the *DEoptim* optimization was in the following format:

a) A first object called *"optim":* inside this element were contained the information about the best result (*ME* and parameter) of the entire calibration together with the number of generations that have been necessary to reach such result.
b) A second object called *"member":* inside this second element are contained many useful information and data about the different intermediate results obtained through the calibration. The best intermediate results (*ME* and parameters) found at each generation are saved and recoverable.

This additional aspect must be clarified: the *HydroPASS* algorithm need not only a single set of parameters for each basin, but a higher number of possible solutions, possibly with the highest possible *ME*. Thankfully inside *"member"* all the needed information is already available. In general, we decided to utilize at least 30 sets of parameters for each catchment possibly with a *ME* greater than 0.7. Now three possible situations presented:

1. After the calibration has ended, at least 30 generations had been evaluated for the considered catchment and all of them had as best individual of the population a set of parameters with *ME* sufficiently high. This is the best-case scenario where no further steps were needed.
2. After the calibration has ended, at least 30 generations had been evaluated but less than 30 presented as best individual a set of parameters with sufficiently high *ME.* Here the adopted strategy has been to run a second time (or more if necessary) the calibration with the same settings. Remember in fact that the *DE* algorithm is stochastic and not deterministic; in other words, starting from the same initial conditions, the results of the calibration are always different and so utilizable without any risk of repetitions.
3. After the calibration has ended, the best solution presented a very poor *ME.* In all these cases the same procedure as in case 2 has been applied but with scarce results. Despite the possibility that the first calibration remained stuck inside a local minimum far from the global one, also in all the successive attempts the result never improved substantially. These basins will be discarded in the preparation of the *HydroPASS* algorithm.

A map showing the spatial distribution of the 117 calibrated basin with their relative performance is reported in Figure 16:

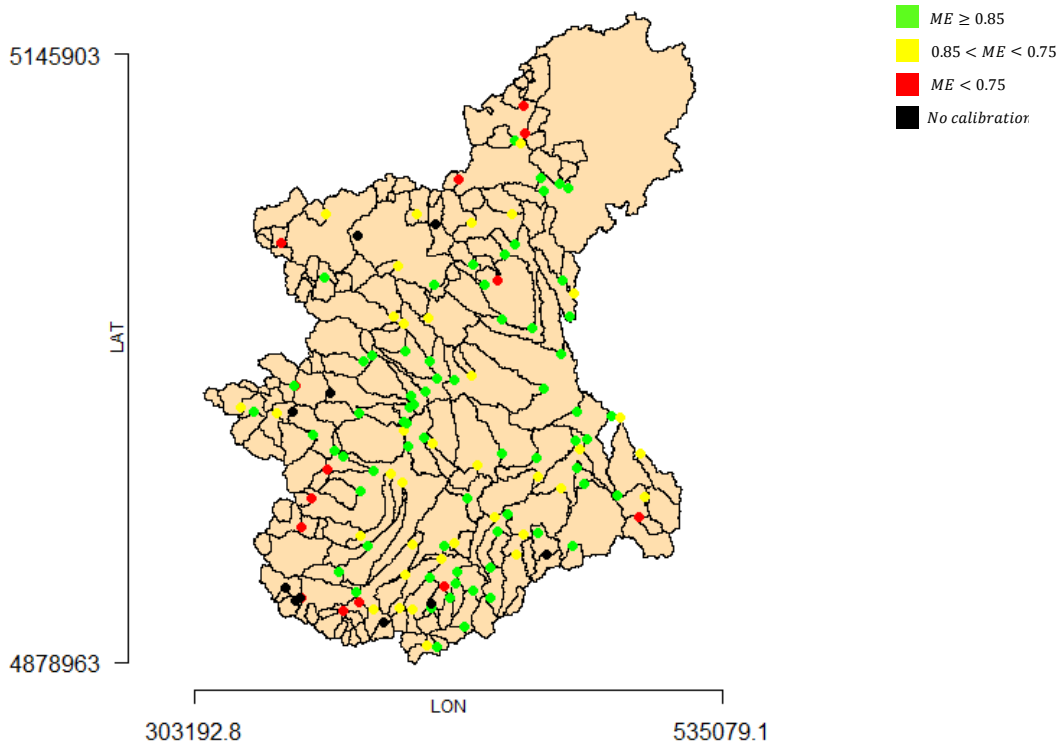


Figure 16 - ME distribution for the best lumped calibration of the catchments

Just from a visual inspection some hypothesis about the low efficiently calibrated basins can be proposed: most of such catchments are in mountainous areas along the Alps or south along the Apennines. The reduced *ME* can be probably related to the high dominance of snow in the hydrological processes; a different objective function also accounting for how good the model is or not to reconstruct snow cover (like the one proposed in $eq.4.12$) could probably improve performance in these locations. Regarding instead the mid value basins there is not an immediate clear possible cause: maybe is just a matter of the model or maybe it can be related to the observed discharge, in some cases limited to few years or with long gaps between two continuous and complete time periods. Nevertheless, overall, we can observe how most of the catchments show a good calibration, for sure inside the moderate or good range values. A global indication of the distribution of the *ME* is instead provided in Figure 17 with the *eCDF* (empirical cumulative distribution function) and the relative boxplot:

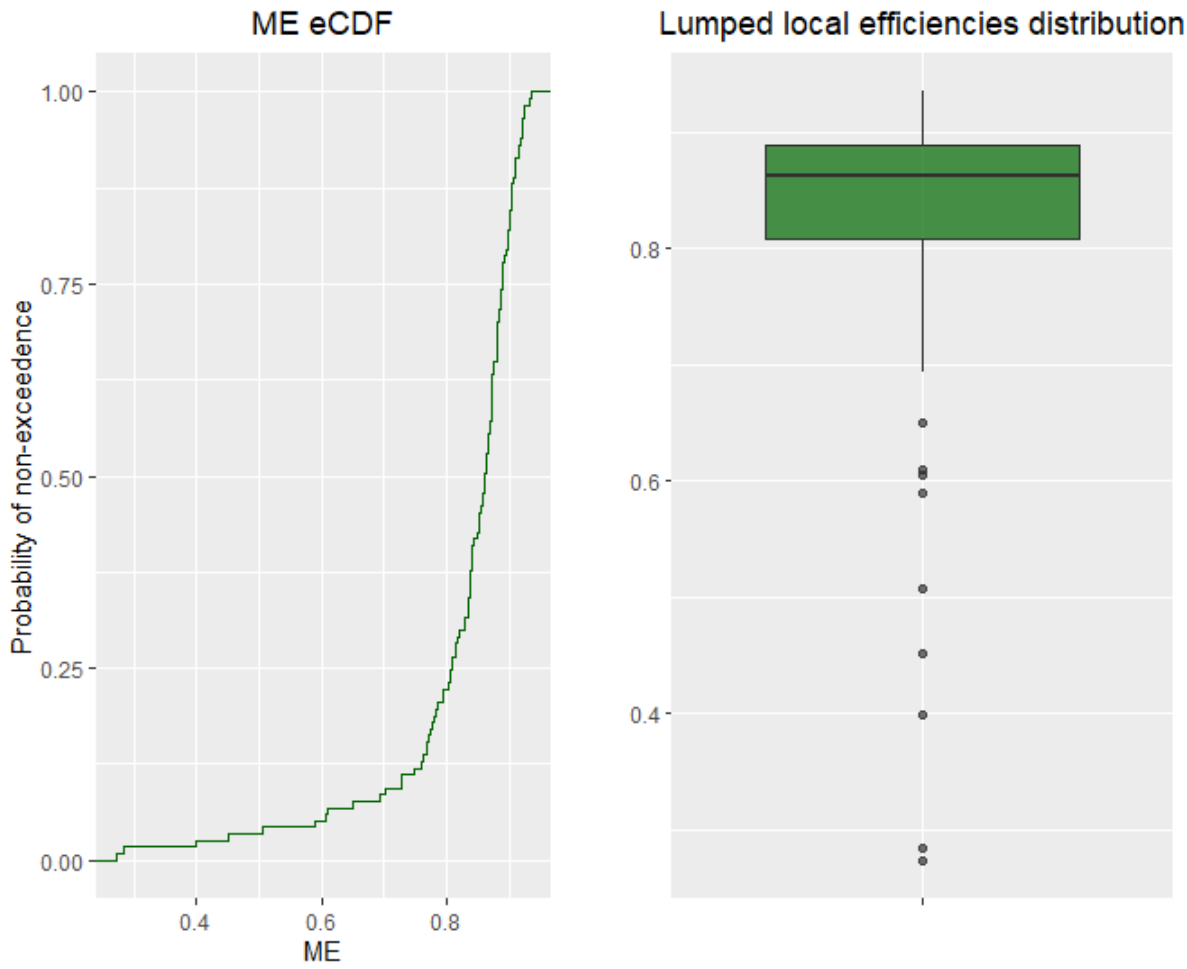


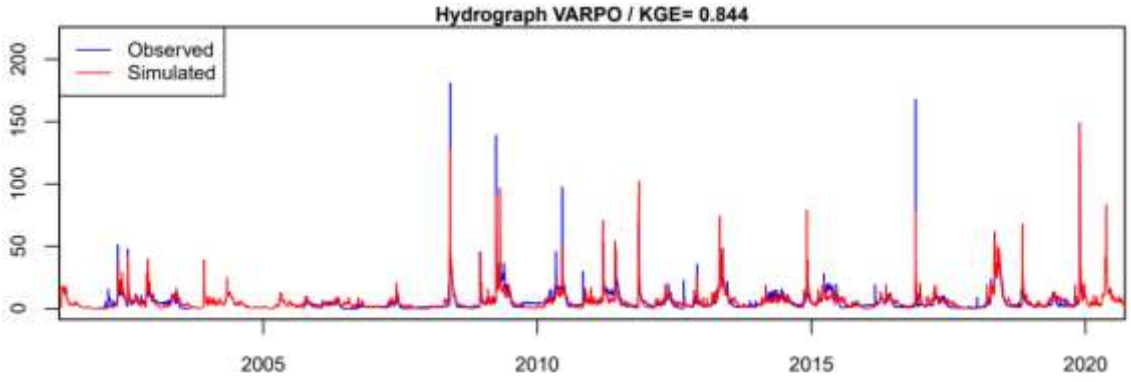
Figure 17 - Empirical cumulative distribution function and boxplot of the lumped ME

The distribution of the lumped *MEs* is generally good, with a mean around 0.86, the 25-percentile around 0.8 and the 75-percentile around 0.89. Moreover, from the boxplot is again visible how the low calibration values are considered outliers, supporting the idea that they are present because they belong to a different distribution, like a subgroup of snow dominated catchments.

4.6 Additional analysis on the simulated discharge and lumped parameter sets

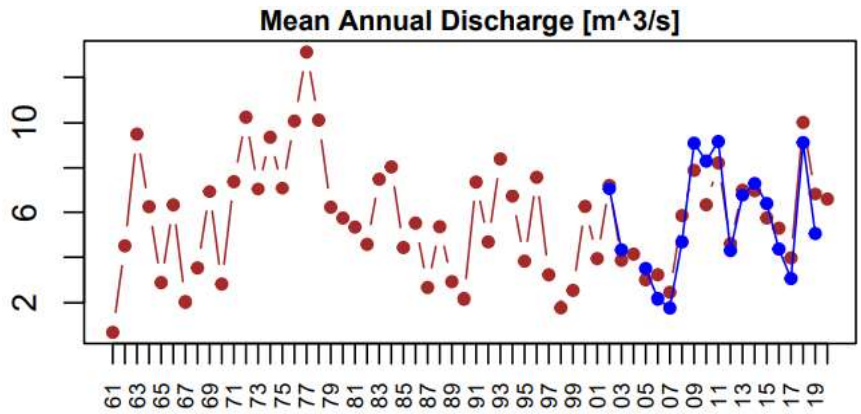
The *ME* is a single number that synthetize the performance of the simulated discharge series with respect to the observed one, but a wider overview of the ability of the model to reconstruct the original data can be explored with some additional analysis. For all the 117 calibrated catchments a *"card"* has been created containing all these additional information. For reference are now reported a brief description of the methods and assumptions made to obtain the graphs for the VARPO basin. A full collection for all the basins is made available in Attachment 1.

A) <u>HYDROGRPH OF THE OBSERVED AND SIMULATED DISCHARGE</u>



A simple visual comparison between the observed and simulated discharge (m^3/s) as obtained applying in lumped form *TUWmodel* with the best available set of parameters. The simulated series is always complete (60 years), while the observed one is of course limited to the available data. Here the timeframe reported is always the one of the observed timeseries. Overall, the match is quite good as expected by a moderately high model efficiency of 0.84, also during the peaks of discharge in correspondence to flood events. Nevertheless, it must be considered how for the basins with lower ME the matching is, reasonably, worst.

B) MEAN ANNUAL DISCHARGE



The mean annual discharge is simply calculated as the average flow at the monitored outlet over a year. Here are reported for the full 60-years' timeframe in m^3/s. Here is considered the calendar year, but another very diffuse approach is instead to consider hydrological years (in Piedmont it starts at the beginning of October and ends at the end of September).

$$\overline{Q_{y,i}} = \frac{\sum_{j=1}^{365} Q_{j,i}}{365} \quad (eq.4.13)$$

Where i indexes the specific year and j the specific day of the year. For the simulated discharge the computation of the average was never a problem being the series always complete and without missing values. For the observed series some specifications are instead needed: a year has been included in the analysis only if the number of missing values (*NA*) was lower than 65 (or in other word if at least 300 days of observation were available for the year considered). Otherwise, the mean has not being evaluated because considered possibly not representative of the real annual average conditions. It was a common condition at the beginning and the end of the monitoring period just like in the middle whenever any interruption of the data recording was present. Overall, being here analyzed a mean statistic and not one related to the extremes, it can be observed how the reconstructed timeseries fully manage to recreate a reliable model of the real field conditions. To give a better idea of the global performances across all the catchments some simple boxplots have been produced:

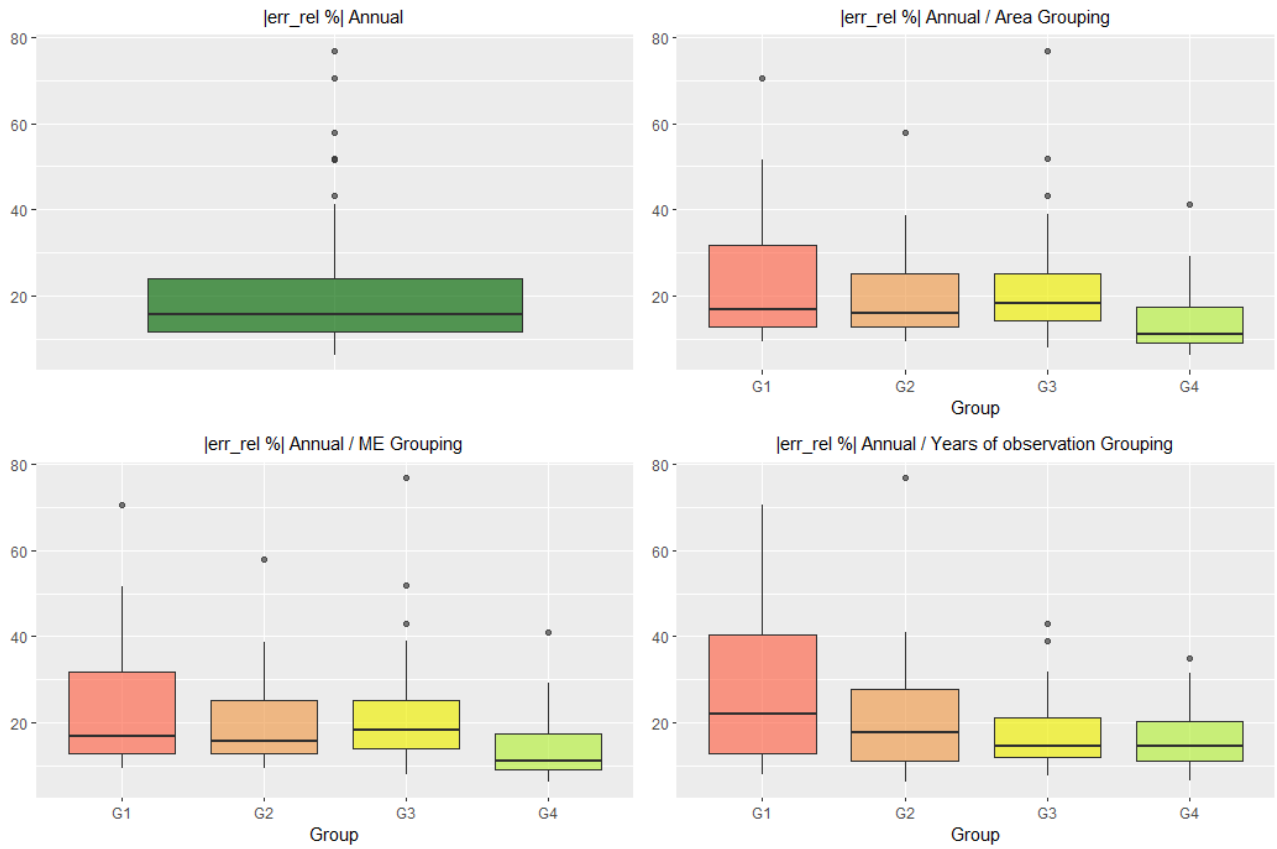


Figure 18 - Boxplot of the average absolute relative error for the annual mean discharge. All catchments (top-right), by catchments area (top-left), by available years of observation (bottom-right), by maximum ME (bottom-left)

The box plots have been constructed considering the average of absolute values of the relative errors between the observed averages and the simulated ones for each i-year and each j-catchment:

$$\left|err_{rel_{i,j}}\right| = \frac{\left|Q_{a,sim,i,j} - Q_{a,obs,i,j}\right|}{Q_{a,obs,i,j}} * 100 \quad (eq. 4.14)$$

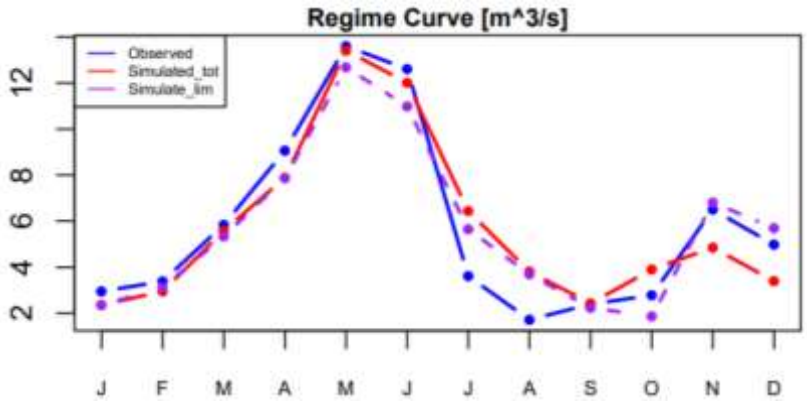
The empty dots represent outliers, the thick line the median, the extremes of the plot the first and third quantiles and the upper and lower lines the maximum and the minimum. The first one (top-left of the image) considers all the catchments together reporting a median around 15%. All the other plots subdivide the basins into categories based on three different criteria as reported in the Table 3:

CRITERION	GROUP 1	GROUP 2	GROUP 3	GROUP 4
CATCHMENT AREA	$A \leq 126\ km^2$ (29)	$126\ km^2 < A \leq 250\ km^2$ (29)	$250\ km^2 < A \leq 801\ km^2$ (29)	$A > 801\ km^2$ (30)
BEST MODEL EFFICIENCY	$ME \leq 0.8$ (26)	$0.8 < ME \leq 0.85$ (24)	$0.85 < ME \leq 0.89$ (39)	$ME > 0.89$ (28)
AVAILABLE YEARS OF OBSERVATIONS	$N° \leq 10$ (20)	$10 < N° \leq 16$ (30)	$16 < N° \leq 18$ (37)	$N° > 18$ (30)

Table 3 – Catchments subdivision based on three criteria (in parenthesis number of elements per group)

The extremes for the intervals in each categorization have been selected to obtain, as far as possible, equal populated classes. In general, it can be observed some clear trends: looking at the catchment area the median is nearly the same in all groups (just slightly lower for the biggest basins), while the spread around the mid value clearly decreases for larger catchments. Regarding the years of observations, a similar trend can be identified, lowering the median (more evidently than before) and reducing the spread with the increase of the data availability; last, looking at the *ME,* the results are the most obvious, showing a clear improvement of the model performance in reconstruction the annual averages moving towards higher values.

C) *REGIME CURVES*



Again, also in this case is analyzed a mean statistic, but now over a seasonal timescale. The regime curve is defined as the set of monthly averages $[m^3/s]$, each computed over the entire observation period. To obtain a meaningful result, possibly a high number of years of measurements are needed (around 30); in hydrology, unfortunately, having such long timeseries of discharge is not so common. In this case, for the first time, in the graph are represented three

curves and not just two. The blue one is related to the observed data, the red one ("$simulated_{tot}$") is related to the simulated series over the full 60-year period; this means that also years where there were not actually observations have been introduced in the calculation. Lastly the purple one is instead still related to the simulated discharge but limited to the years overlapped with the observations:

$$\overline{Q_{m,\iota}} = \frac{\sum_{j=1}^{k} Q_{i,k}}{k} \quad (eq.\,4.15)$$

Whit i indexing the month, j the day of the month and $m = 28{,}29{,}30{,}31$ depending on the month and on leap years. It can be easily seen, and it is a common feature across all the catchments, how the total simulated series apparently does not reproduce correctly the monthly hydrological behavior, but when the timeframe is reduced, the matching improves significantly. Again, overall, the correspondence seems quite good, even if slightly worse than for the annual averages. Also, in this case a wider overview for all the catchments can be provided looking at some additional boxplots:

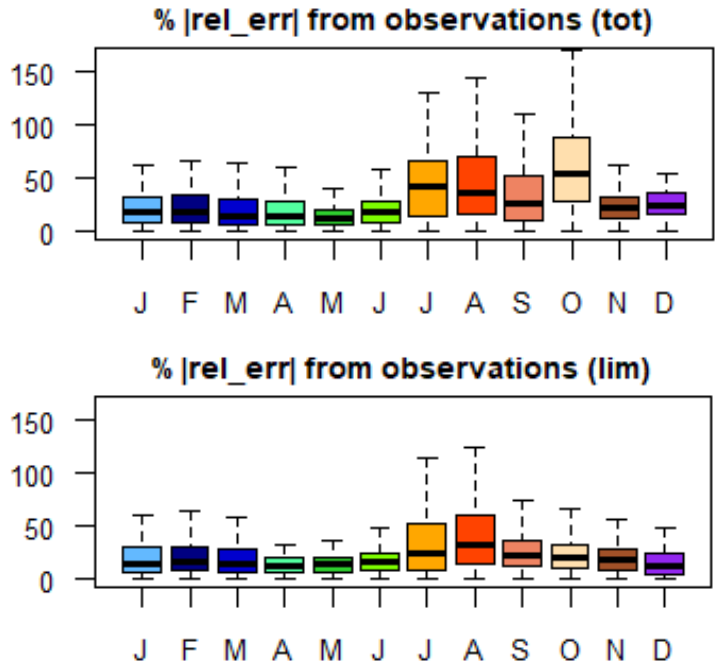


Figure 19 - Boxplot (all catchments) of the average absolute relative error of the monthly discharge between observed and total simulated (above) and limited simulated (below) discharge

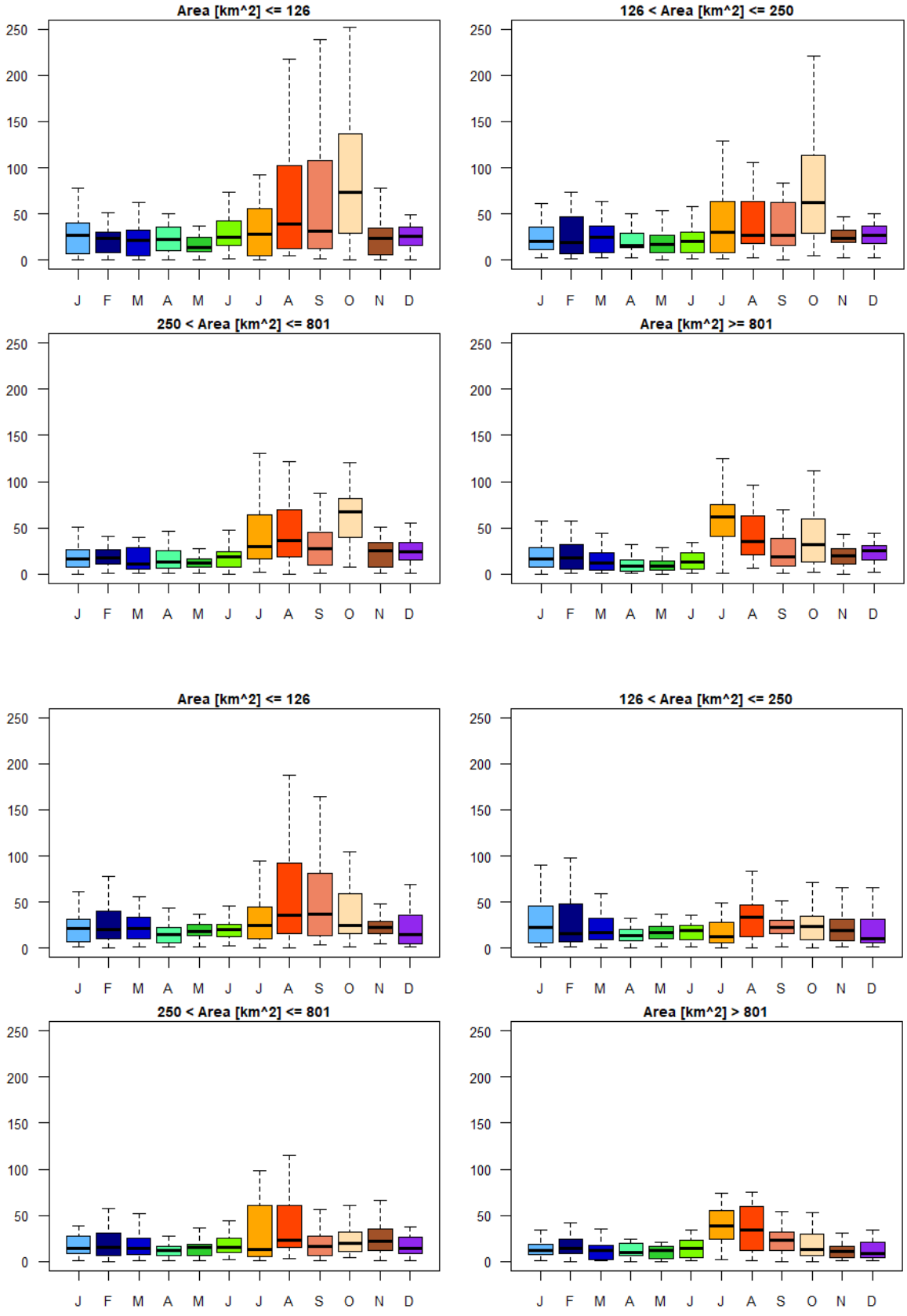


Figure 20 - Boxplots of the average relative error of the monthly discharge between the observed and total simulated (above) and limited simulated (bottom) discharge [Catchment Area]

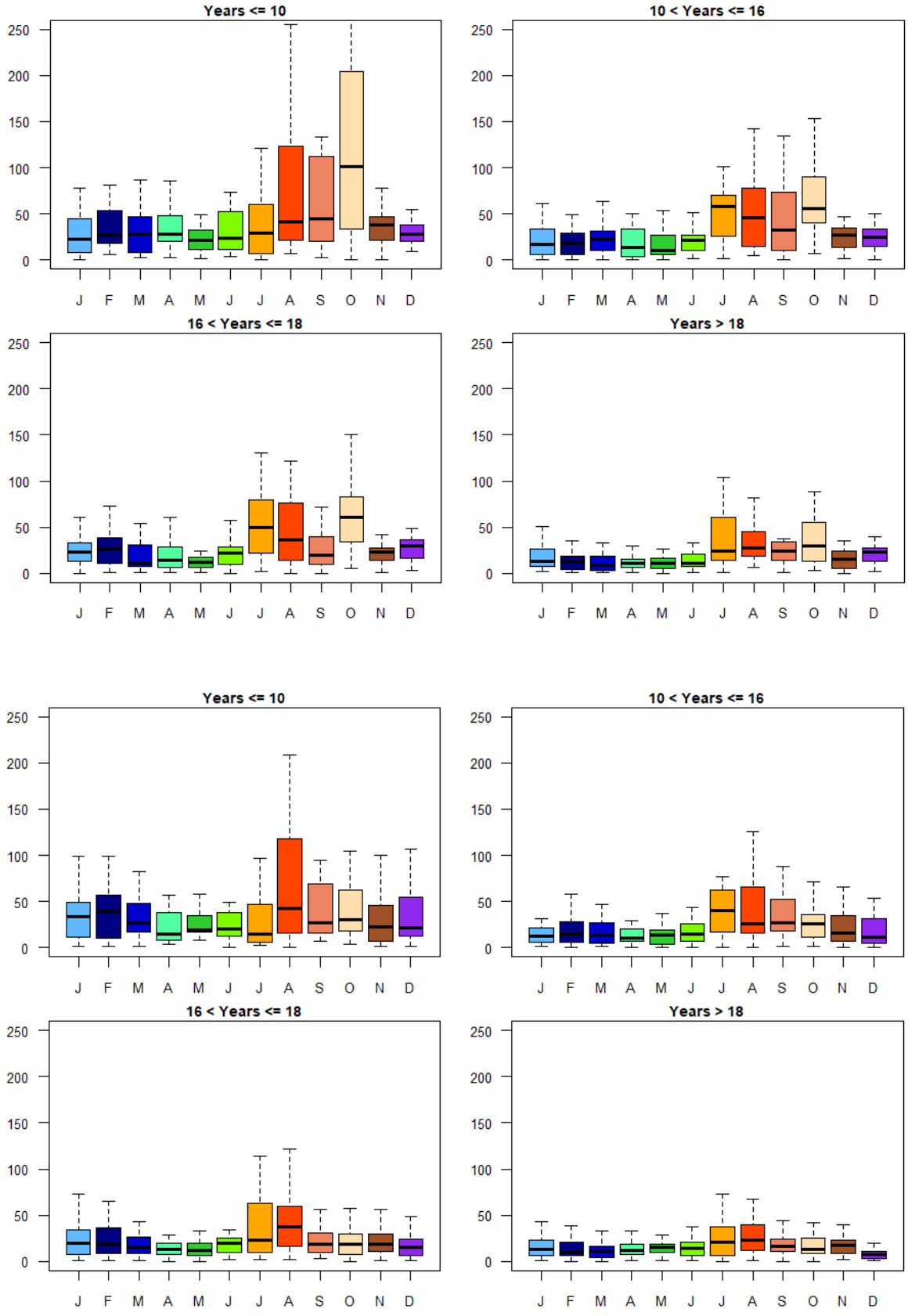


Figure 21 - Boxplots of the average relative error of the monthly discharge between the observed and total simulated (above) and limited simulated (bottom) discharge [Years of observation]

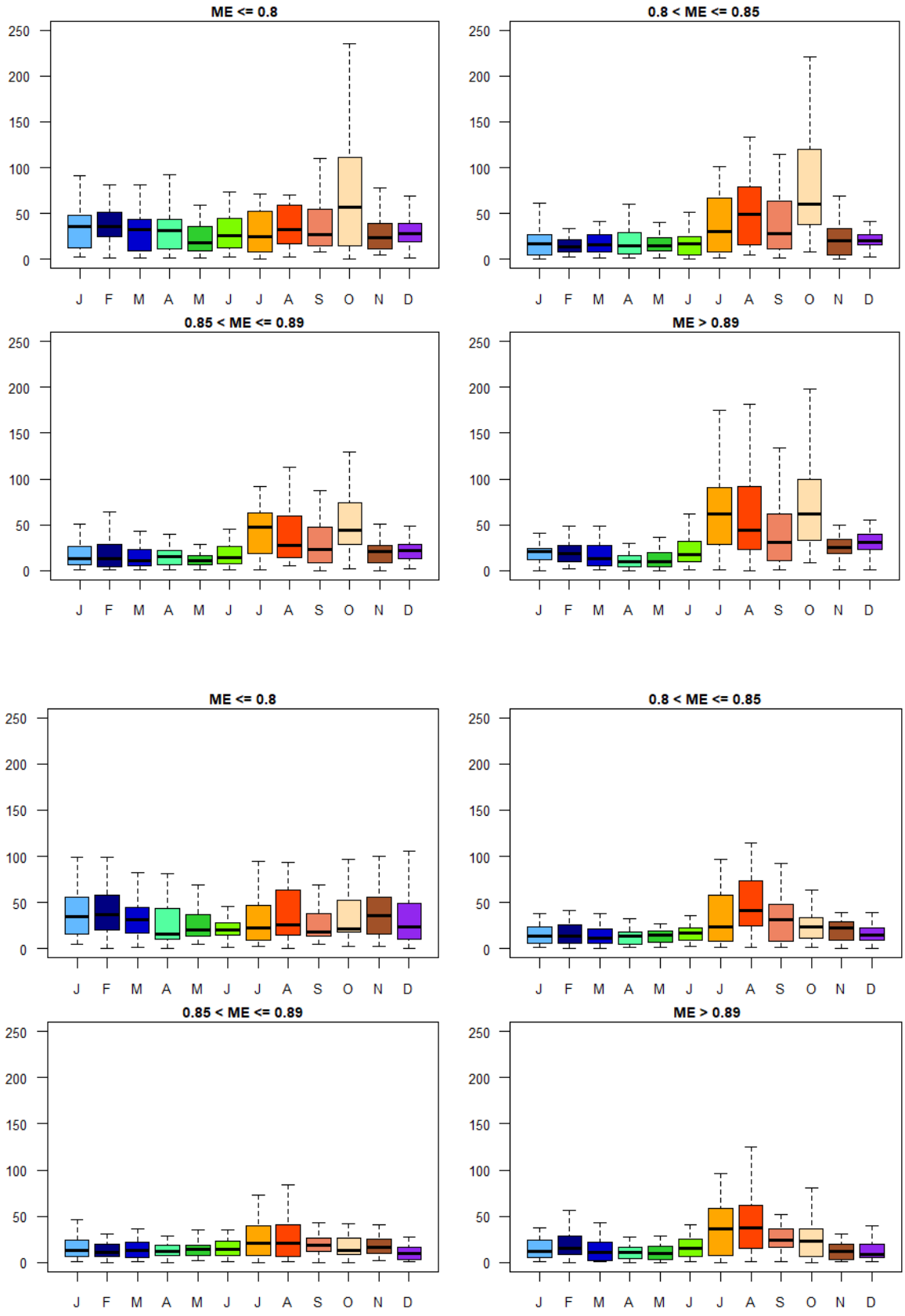
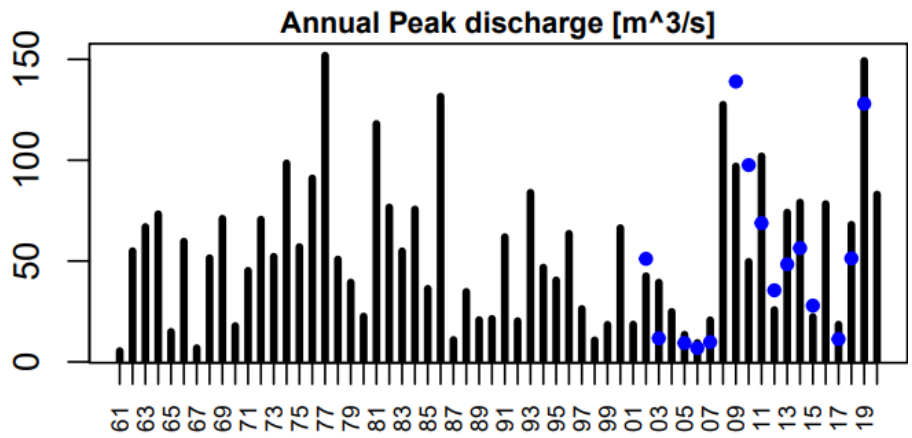


Figure 22 - Boxplots of the average relative error of the monthly discharge between the observed and total simulated (above) and limited simulated (bottom) discharge [Model Efficiency]

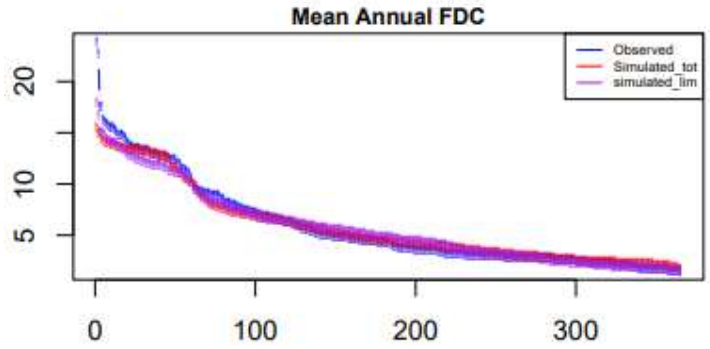
These graphs are subdivided, as the previously showed for the mean annual discharge, in the three groups (Figures 20-21-22), plus the one showing the overall statistic of the catchments all together (Figure 19). Additionally, the analysis has been divided for each month to better represent the seasonal response of the model. On one hand the same general trends as already presented before for the three criteria are present, with an improved performance increasing the area of the basins, the number of available years of data and the *ME*. On the other hand, another trend is apparently present: the months from June to October are seemingly worst reconstructed (higher median of the relative percentual differences) and present a wider spread around the mid value. This trend is more evident where it is additionally combined with one of the unfavorable grouping conditions. It is not the goal of this thesis to investigate the reasons behind such model behavior, but an improvement in this direction (modifying something inside the model itself) could reduce the differences, increasing the *ME* and, overall, the performance of the following *HydroPASS* algorithm. As usual the limited simulated discharge timeseries obtain a better match than the 60-years ones.

D) *ANNUAL PEAK DISCHARGE*



It a simple comparison between the highest daily discharge recorded in the observed data and what in the same years has been simulated with *TUWmodel* $[m^3/s]$. Talking about extremes (that are intrinsically more difficult to reproduce through a model) the matching is evidently worst, but nevertheless there is still accordance for most of the years: when a peak flow has been recorded in the original data, a peak is generally present also in the simulated series, even if not necessarily with the same magnitude (but nevertheless there is a common reconstructed trend).

E) *MEAN ANNUAL FLOW DURATION CURVE (FDC)*



A *FDC* (flow duration curve) is a curve that represents the percentage of times that a given discharge value is exceeded. In one way is very similar conceptually to a *CDF* (cumulative distribution function) that instead represents the probability of non-exceedance. To make an example in the graph above for a value of $10\ m^3/s$ we can read on the horizontal axis a value of around 65 days, meaning that on average we expect during a representative year to see a flow rate greater than $10\ m^3/s$ for 65 days. On the x-axis here is not reported properly a probability, but the number of days in a year. An equivalent curve directly expressing the probability of exceedance can be obtained simply by normalizing the values on the horizontal axis by 365 (ranging in this way from 0 to 1). In this specific case is reported a mean annual flow duration curve, just one of the possible *FDCs*: it is constructed starting from the *annual flow duration curves* of the single individual years. An annual flow duration curve is obtained simply by ordering in descending order all the discharged observed in a year. Let us make a concrete example: for the VARPO catchment data of the years from 2002 to 2020 were available; for each of the 19 years an annual *FDC* is constructed by ordering the observed discharge values in decreasing order. In this way ordered series of 365 measurements are obtained. The mean annual *FDC* for the basin is obtained by averaging the first values of each curve, the second values and so on till the 365^{th} values. The same exact reasoning can be done for the simulated timeseries, again distinct into the "total" series (60 years) and the "limited" series (same number of years of the observation). It must be specified how while calculating the annual *FDC* of the individual years, in case more than 65 missing (*NA*) or non-valid values were present, the entire year has been neglected and taken out from the mean computation. Similarly, to what already observed for the regime curves (even if in the specific case of the VARPO basin in not that evident), the limited timeseries is able to

better approximate, in most cases with good results, the signature of the observed discharge. The total simulated timeseries struggles instead more.

F) *PARAMETERS SETS COMPARISON /EQUIFINALITY*

The last analysis about the model performance has been conducted on the parameters sets found for each catchment. As already described, 30 parameters sets have been obtained for each basin to be used as input element for the *HydroPASS* algorithm.

The graphs reported in Figures 23 and 24 have been produced to assess the issue of model parameters uncertainty and equifinality. *"The term equifinality was used for the first time in the field of hydrological modelling by Beven (1975) who lately proposed in 1993 a concept of equifinality for model evaluation and uncertainty analysis" (Equifinality and Flux Mapping: A New Approach to Model Evaluation and Process Representation Under Uncertainty, Sina Khatami et al.,* 2019*).* Within hydrological literature parameter equifinality and uncertainty are treated similarly and interchangeably. Parameter uncertainty means that there is no certain/true parameter set and it is eventually represented probabilistically as parameter distributions. More than one parameter set can be accepted and can result in a similar *ME*. This uncertainty derives from the intrinsic ill posedness and ill conditionedness of environmental models (Beck, 2002; Beven, 2006). Generally, parameters uncertainty is investigated by searching multiple valid sets by meaning of, for example, Monte-Carlo simulations. In our case, considering the already availability of 30 sets per basin, the equifinality can be investigated without any further computations. Below are reported two graphs: the first shows the best 10 normalized parameter sets (highest *ME*) with the indication of the mean model efficiency obtained considering the best 20 out of 30 sets. The normalization of each individual model parameter has been done following the relation:

$$P_{i,j,k,norm} = \frac{P_{i,j,k} - P_{i,min}}{P_{i.max} - P_{i,min}} * MAX_{scale} \quad (eq.4.16)$$

Where $P_{i,j,k,norm}$ is the i-normalized parameter (i ranges between 1 and 15) of the j-set of the k-catchment, $P_{i,j,k}$ is the original parameter, $P_{i,min}$ and $P_{i.max}$ are the minimum and maximum boundary of the i-parameter (see Table 2) and MAX_{scale} represents the maximum value the new normalized parameter can assume in the new scale (from 0 to $MAX_{scale} = 1$ in this specific case). In other words, the closer the new normalized parameter is close to zero, the closer is the inferior

limit of the possible admissible values, the closer it is to one the closer it is instead to the upper limit:

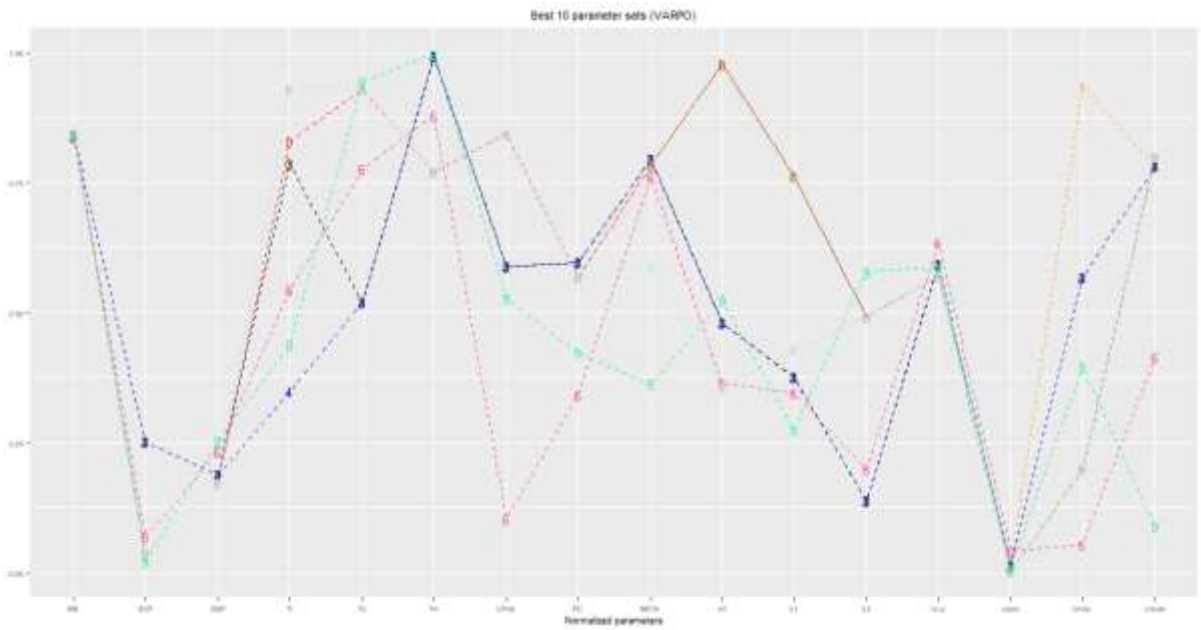


Figure 23 - Normalized Parameter sets (VARPO)

It can be noticed how the different sets, despite being all valid and with extremely similar *ME* (for the VARPO basin are nearly coincident, in other cases not necessarily), does not seem to concentrate consistently along precise values (even if small subsets of parameters seem to follow very similar parameterizations). Seemingly, the parameters, if opportunely combined, are always able in some way to well perform and reconstruct the observed discharge. A different trend will be observed after the application of the HydroPASS algorithm (see chapter 6). Moreover, in some cases the parameters are extremely close, if not coincident, with the upper and lower limits; this is probably due to the *DE* algorithm strategy to correct, after vector mutation, the value exceeding the allowed boundaries ($eq.\,4.9a$ and $eq.\,4.9b$).

The second graph reports instead for the 15 model parameters their probability distribution, using all the 30 available sets, through boxplots:

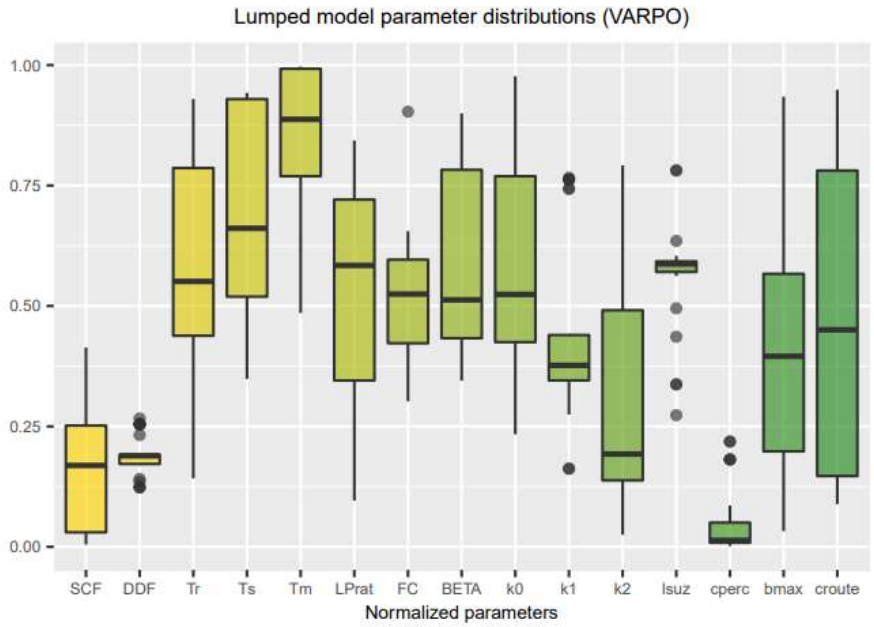
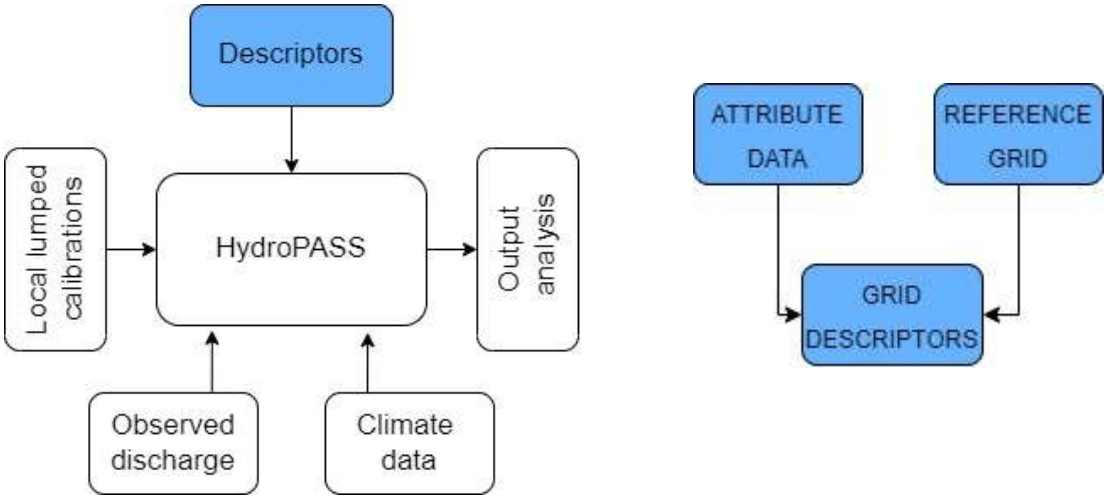


Figure 24 - Local lumped parameter distributions (VARPO)

In this way is clearer how the equifinality is very strong for some parameters like $Tr, Ts, Lprat$ or $croute$, while for other (especially $DDF, lsuz$ and $cperc$) the value concentrated around a specific value. For these last cases it seems that a stronger correlation, underlying a clear physical relationship, is present (at least for the VARPO catchment). The same cannot be stated for the remaining parameters. Moreover, other basins, despite showing a similar behavior, may do it with different parameters. For a full compendium of all the other catchments see Attachments 2 and 3.

5. CATCHMENT DESCRIPTORS



5.1 Descriptors selection

The last remaining input element to the HydroPASS algorithm still to be discussed are the descriptors. A physical descriptor, that here will be differently denoted in case it is referred to a grid pixel or to a specific basin, is a single number (dimensional or dimensionless) that synthesize a facet of the territory. Other interchangeable terms that can be found in literature are characteristics or attributes (*"Technical research report - volume IV - physical catchment descriptors"*, Paul Mills, Oliver Nicholson and Duncan Reed, 2014). For simplicity in the following paragraphs the descriptors will be denoted simply with *CD* (Catchment Descriptors, even if in this paragraph will be only calculated over the reference grid pixels). The great advantage of a *CD* is the possibility to summarize in a simple and direct way features sometimes rather complex; this gave us indications about the difference between pixels or catchments upon which, in the following chapter, the application of the machine learning technique will be based.

In literature there are a multitude of different *CDs* that can be calculated, and the list could go on forever. A choice of a reduced number of *CDs* was needed: the selection of the set of *CDs* has been based, when possible, given the available data, on what *Merz et al.* did in their original article published in 2020. In the end, a subset of that original list has been here adopted, plus some additional *CDs* not originally present, as reported in the Table 4:

GROUP	LABEL	UNITS	DESCRIPTION
CLIMATE	CL_MAP (1)	mm	Long-term mean-annual precipitation
	CL_MAT (2)	$°C$	Long-term mean-annual temperature
	CL_PET (3)	mm	Long-term mean-annual evapotranspiration
	CL_PETovP (4)	–	Aridity index
	CL_Psum2win (5)	–	Ratio of long-term summer precipitation and winter precipitation
	CL_R50 (6)	mm/day	Long-term median maximum daily precipitation
	CL_R95 (7)	mm/day	Long-term 95th percentile of maximum daily precipitation
	CL_dRD2D (8)	mm	Long-term mean absolute difference of rainfall amount between two consecutive days
MORPHOLOGY AND TOPOGRAPHY	MP_mean_dem (9)	$m\ asl$	Mean elevation
	MP_CV_dem (10)	–	Coefficient of variation of elevation in the catchment
	MP_mean_slope (11)	%	Mean slope
	MP_mean_aspect (12)	°	Mean aspect
LAND USE	LD_smallveg (13)	%	Percent of the pixel covered with herbaceous, little or no vegetation, and open spaces
	LD_agri (14)	%	Percent of the pixel covered with agricultural areas
	LD_wetland (15)	%	Percent of the pixel covered with wetlands and lakes
	LD_urban (16)	%	Percent of the pixel covered with artificial surfaces
	LD_forest (17)	%	Percent of the pixel covered with various types of forests
CURVE NUMBERS	CN1 (18)	–	Curve number 1
	CN2 (19)	–	Curve number 2
	CN3 (20)	–	Curve number 3
SOIL PHYSICAL AND WATER PROPERTIES	S_SILT (21)	%	Mean fraction of silt in subsoil (30-100 cm)
	T_SILT (22)	%	Mean fraction of silt in topsoil (0-30 cm)
	S_SAND (23)	%	Mean fraction of sand in subsoil (30-100 cm)
	T_SAND (24)	%	Mean fraction of sand in topsoil (0-30 cm)
	S_CLAY (25)	%	Mean fraction of clay in subsoil (30-100 cm)
	T_CLAY (26)	%	Mean fraction of clay in topsoil (0-30 cm)

	S_GRAVEL (27)	%	Mean fraction of gravel in subsoil (30-100 cm)
	T_GRAVEL (28)	%	Mean fraction of gravel in topsoil (0-30 cm)
	S_REF_BULK_DENSITY (29)	kg/dm^3	Mean Bulk Density of subsoil (30-100 cm)
	T_REF_BULK_DENSITY (30)	kg/dm^3	Mean Bulk Density of topsoil (30-100 cm)
	AWC_LARGE (31)	%	Percent of pixel with large (125-150 mm/m) available water content (*FAO,* 2006)
	AWC_MED (32)	%	Percent of pixel with medium (75-125 mm/m) available water content (*FAO,* 2006)
	AWC_SMALL (33)	%	Percent of pixel with small (15-75 mm/m) available water content (*FAO,* 2006)
	IL_TOP (34)	%	Percent of pixel with impermeable layer located within 80 cm of soil profile
	IL_MED (35)	%	Percent of pixel with impermeable layer located within 80-150 cm of soil profile
	IL_DEEP (36)	%	Percent of pixel with no impermeable layer located within 150 cm of soil profile
	SWR_NOTWET (37)	%	Percent of pixel with dominant annual average soil water regime class: not wet within 80 cm for over 3 months and not wet within 40 cm for over 1 month
	SWR_MEDIUMWET (38)	%	Percent of pixel with dominant annual average soil water regime class: wet within 80 cm for 3 -6 months, but not wet within 40 cm for over 1 month
	SWR_WET (39)	%	Percent of pixel with dominant annual average soil water regime class: wet within 80 cm for over 6 months, but not wet within 40 cm for over 11 months
	SWR_TOTALWET (40)	%	Percent of pixel with dominant annual average soil water regime class: wet within 40 cm for over 11 months
	T_TEXTURE_COARSE (41)	%	Percent of pixel with coarse topsoil texture
	T_TEXTURE_MEDIUM (42)	%	Percent of pixel with medium topsoil texture
	T_TEXTURE_FINE (43)	%	Percent of pixel with fine topsoil texture
	S_USDA_CLAY (44)	%	Percent of pixel with clay subsoil according to USDA classification
	T_USDA_CLAY (45)	%	Percent of pixel with clay topsoil according to USDA classification

	S_USDA_SAND (46)	%	Percent of pixel with sand subsoil according to USDA classification
	T_USDA_SAND (47)	%	Percent of pixel with sand topsoil according to USDA classification
	S_USDA_SILTLOAM (48)	%	Percent of pixel with silt and loam subsoil according to USDA classification
	T_USDA_SILTLOAM (49)	%	Percent of pixel with silt and loam topsoil according to USDA classification
	DRAINAGE_LARGE (50)	%	Percent of pixel belonging to “excessive” and “well” drainage class. Soil drainage classes are based on the guidelines from *FAO* (2006):
	DRAINAGE_MED (51)	%	Percent of pixel belonging to “moderate” and “imperfect” drainage class. Soil drainage classes are based on the guidelines from *FAO* (2006):
	DRAINAGE_SMALL (52)	%	Percent of pixel belonging to “poor” and “very poor” drainage class. Soil drainage classes are based on the guidelines from *FAO* (2006):

Table 4 - Catchment descriptors (label, units, and description)

Each of the reported *CD* was to be calculated over the reference grid used for the project. Depending on the original data used, case by case it was convenient to work directly in *UTM32* or, like how already done for the climate data, in *WGS84*, referring then the results to the reprojected grid (see paragraph 3.1). The application of one or the other strategy has been used to reduce as much as possible the reprojection of raster files and the introduction of unnecessary interpolations. Starting from the following paragraph, a more detailed overview of the *CDs* (source data and methodology), following the classification reported in Table 4, will be provided.

5.2 Climate descriptors

The first subcategory of *CDs* is the one related to the climate characteristics. For all of them the original data needed were the precipitation (P), temperature (T_{mean}) and evapotranspiration (PET) already presented in paragraphs 3.1 and 3.2**.** All these data were in *WGS84* and so the results have been then referred to the grid in *UTM32* without any need of further interpolations. The *CL_MAP [mm]*, *CL_MAT*

[°C], *CL_PET [mm]* are all calculated following the same approach: they are long term averages of the corresponding climatic variable obtained as:

$$CL\_MAP_i = \overline{P_i} * 365.25 \quad (eq.5.1a)$$
$$CL\_MAT_i = \overline{T_i} * 365.25 \quad (eq.5.1b)$$
$$CL\_PET_i = \overline{PET_i} * 365.25 \quad (eq.5.1c)$$

Where the index i denotes the pixels and the $\bar{x}$ variables the averages over the 60-years reference period (excluding missing values).

The aridity index *CL_PETovP [-] (Budyko,* 1974*)* is defined by the ratio of the mean annual potential evapotranspiration and mean annual precipitation just showed. It is basically the reciprocal of the aridity index as adopted in 1992 by the *UNEP* (united nations environment programme).

$$CL_{PETovP_i} = \frac{CL\_PET_i}{CL\_MAP_i} \quad (eq.5.2)$$

The Budyko index is a measure of the dryness of an environment. Values lower than 1 indicates an energy-limited system (water availability but not enough energy to evapotranspire it all), while values higher than 1 on the other hand indicates water-limited systems (enough energy to potentially evapotranspire more water than the available one).

CL_Psum2win quantify the difference between the long-term summer and winter precipitations. It is calculated as the ratio between the two. The summer period at our latitudes extends from May to October, while the winter period from November to April (in a two seasons calendar subdivision):

$$\mathrm{CL}_{\mathrm{Psum2win_i}} = \frac{\sum_{summer} \overline{P_{j,i}}}{\sum_{winter} \overline{P_{k,i}}} \quad (eq.5.3)$$

Where j and k ranges from 1 to 6. Each individual monthly long term precipitation average is calculated with a similar equation to $eq.5.1$.

The long-term median maximum daily precipitation *CL_R50* is calculated considering the maximum daily precipitation for each available year and then applying the median operator:

$$CL_{R50_i} = median\left(P_{1,i}, \dots, P_{j,i}\right) \quad (eq. 5.4)$$

With i denoting the pixel and j, ranging from 1 to 60, the available years of data. *CL_R95* follows a similar approach for the determination of the long-term 95th percentile of maximum daily precipitation; for each pixel the long-term precipitation of each year is evaluated, and the value corresponding to the probability of non-exceedance of 95% is selected (or the same the values corresponding to the probability of exceedance of 5%):

$$CL\_95R_i = 95percentile\left(P_{1,i}, \dots, P_{j,i}\right) \quad (eq. 5.5)$$

Lastly, for the climate descriptors, *CL_dRD2D*. The long-term mean absolute difference of rainfall amount between two consecutive days, as the name suggest, is evaluated by calculating, for each pair of consecutive days, the absolute value of their difference, and then applying the mean operator to the new obtained data series. A simplified explanatory example is here reported:

$$Original \rightarrow 1 \quad 2 \quad 2 \quad 4 \quad 1$$
$$Difference \rightarrow 1 \quad 0 \quad 2 \quad -3$$
$$Absolute\ difference \rightarrow 1 \quad 0 \quad 2 \quad 3 \rightarrow mean = 1.5$$

For each of the eight climate descriptors the corresponding distribution map over the original *OI* grid in *WGS84* is reported in Figures 25-26. Please note that the choice to represent the maps in *WGS84* is only for graphical convenience: remember that the reference system used for the calculation is always *UTM32* (the same also apply for all the other descriptors' classes).

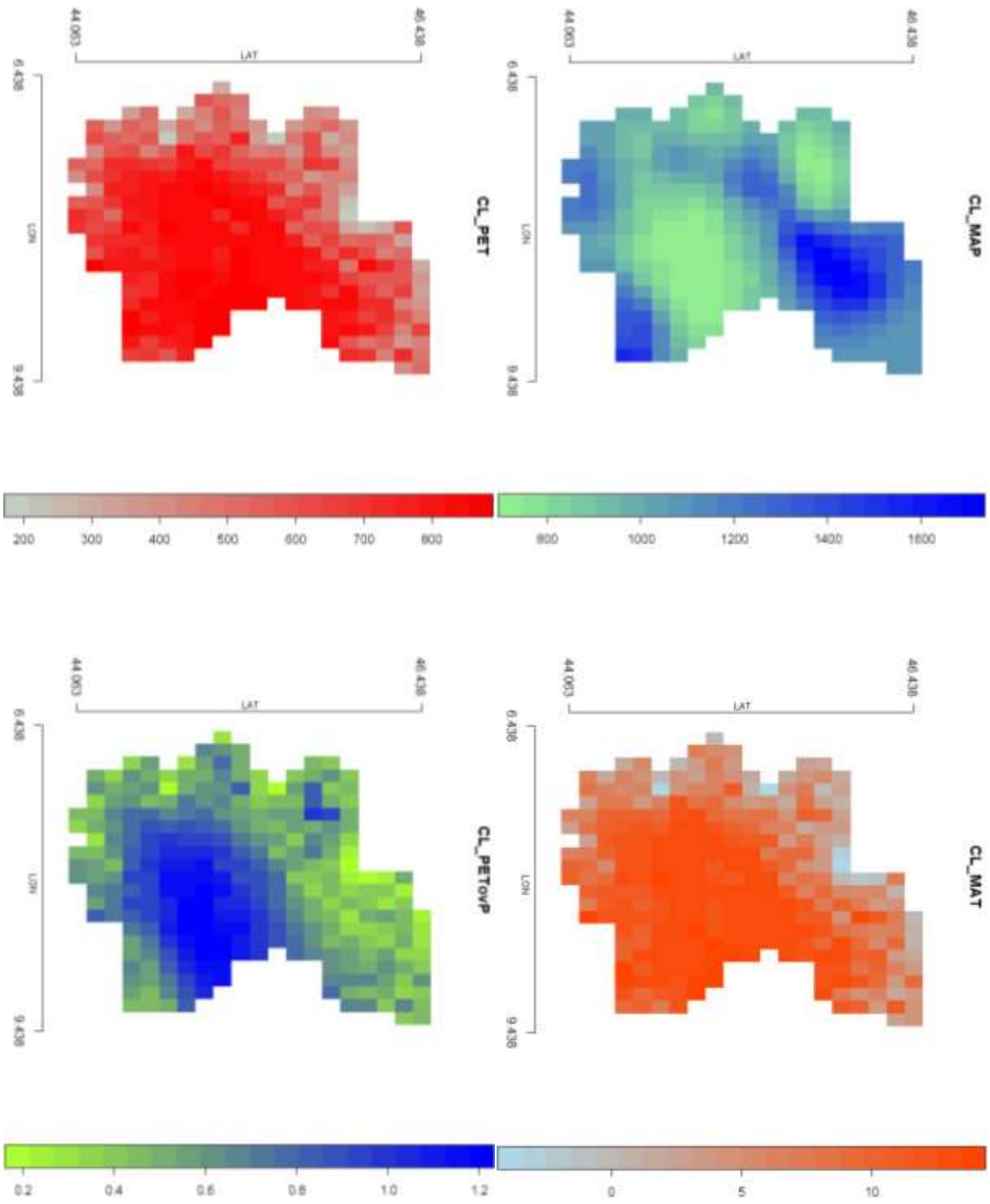


Figure 25 - Spatial map distribution (CLIMATE DESCRIPTORS) of CL_MAP, CL_MAT, CL_PET, CL_PETovP

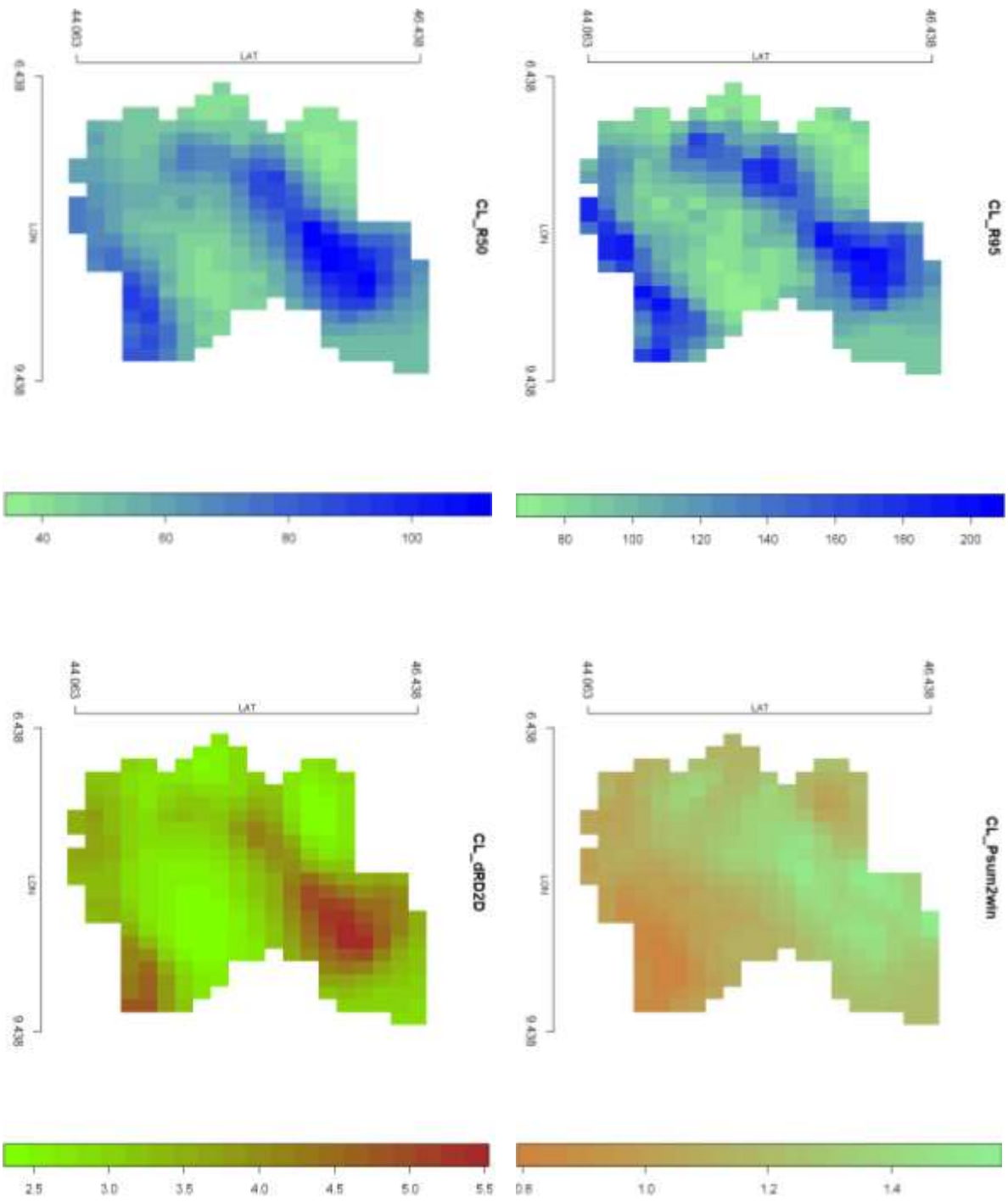


Figure 26 - Spatial map distribution (CLIMATE DESCRIPTORS) of CL_R95, CL_Psum2win, CL_R50, CL_dRD2D

5.3 Morphology and topography descriptors

Four distinct *CDs* belong in this group. The base data to calculate them was a *DTM* (digital terrain model); the one used in this project is a raster file, in coordinate reference system *WGS84*, with squared cells with a resolution of $0.00083°$ (around $90m$ on the ground) downloaded from *ErthEnv-DEM90* digital elevation model (*Robinson at. al,* 2014). The evaluation of *MP_mean_dem* and *MP_CV_dem* followed the same approach just changing the function used; the main difference with respect to the climate *CDs* is that here the resolution of the raster file and of the reference grid are not the same. Specifically, the resolution of the *DTM* is higher than the one of the pixels of the grid, and so in each reference cell of the model fall more than one cell of the raster. The calculation of the descriptors for each cell passed through the determination of the subset of cells of the *DTM* falling into each pixel, then averaging or applying a different function to them. Such operation inside the *R* environment has been easily performed with the dedicated function of the *"raster"* package *"extract"*:

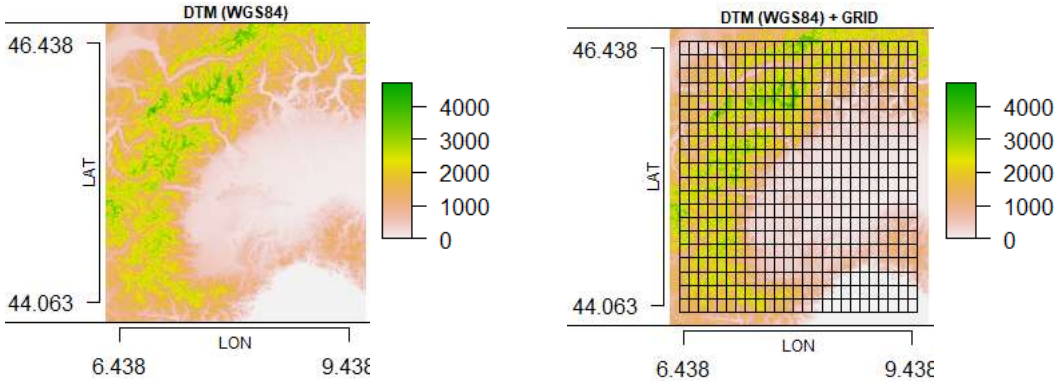


Figure 27 - DTM (90m) with and without reference grid (WGS84)

$$MP\_mean\_dem_i = \frac{\sum_{j=1}^{n} E_{j,i}}{n} \qquad (eq.5.6a)$$

$$MP\_CV\_dem_i = \frac{\sigma_{e,i}}{MP\_mean\_dem_i} \qquad (eq.5.6b)$$

With j indicating the subsets of *DTM* cells for each grid pixel i, and $\sigma_{e,i}$ the standard deviation of the elevation of the pixels. As usual eventual missing values are neglected from the computation. The remaining two descriptors are the one related to the slope and the one related to the aspect. The slope is a measure of the steepness of a digital terrain model, calculated through the elevation differences of adjacent cells; the lower the slope (expressed in $\%$) the flatter the terrain. Positive

or negative values of the slope indicates instead the direction of the terrain inclination. The aspect [°], also called exposure, is instead the measure of the compass direction (or azimuth) that a terrain surface faces. Such calculations can be performed directly inside *R* with the external package *"rgrass7"* that introduces an indirect access to the *GRASS* toolbox of *QGis*. For simplicity has been decided to directly work inside *QGis* instead. The obtained raster files of slope and aspect have then been imported again in *R* to perform (as for the elevation and coefficient of variation) the extraction of the cells subsets and the calculation of the mean values as descriptors:

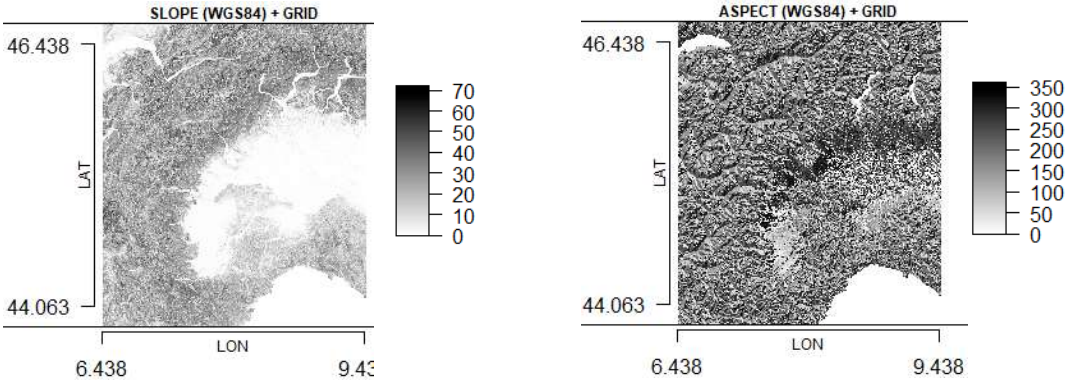


Figure 28 - Slope and Aspect maps (WGS84)

$$MP\_mean\_slope_i = \frac{\sum_{j=1}^{n} SLOPE_{j,i}}{n} \qquad (eq.5.7a)$$

$$MP\_mean\_aspect_i = \frac{\sum_{j=1}^{n} ASPECT_{j,i}}{n} \qquad (eq.5.7b)$$

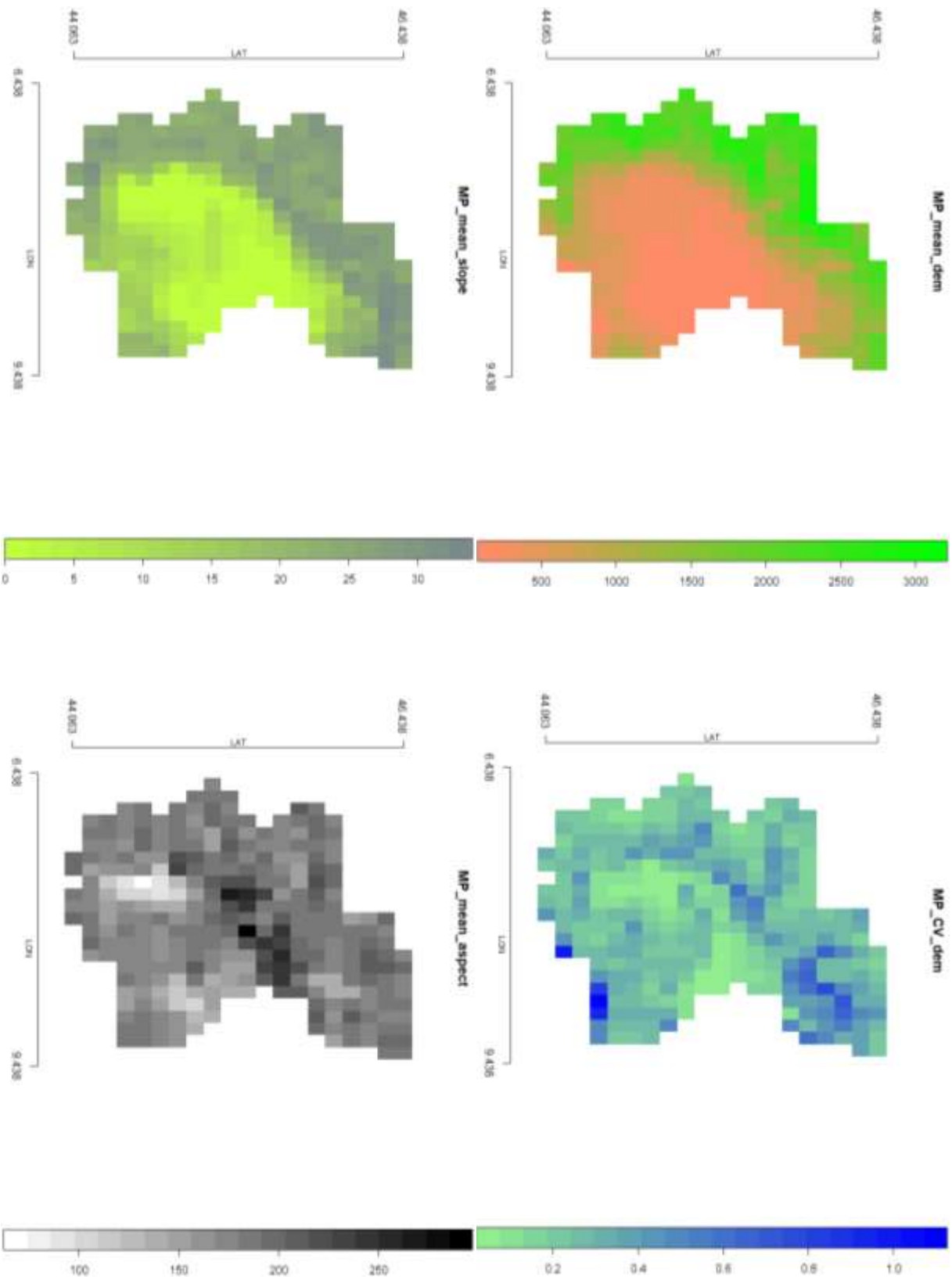


Figure 29 - Spatial map distribution (MORPHOLOGY and TOPOGRAOHY DESCRIPTORS) of MP_mean_dem, MP_CV_dem, MP_mean_slope, MP_mean_aspect

5.4 Land use descriptors

The definition of the five land use descriptors has started from the data contained into the *CORINE* Land Cover (*CLC*) database. Such inverntory started in 1985 and have seen uptadates in the following years till the last version released in 2018 (with a time consitency ranging between 2017 and 2018), here used as data source. In this last version the data have been collected with satellites, specifically Sentinel-2, with the additional use of Landsat-8 for gap filling. The land is subdivided and categorized into 44 classes covering 39 different European nations:

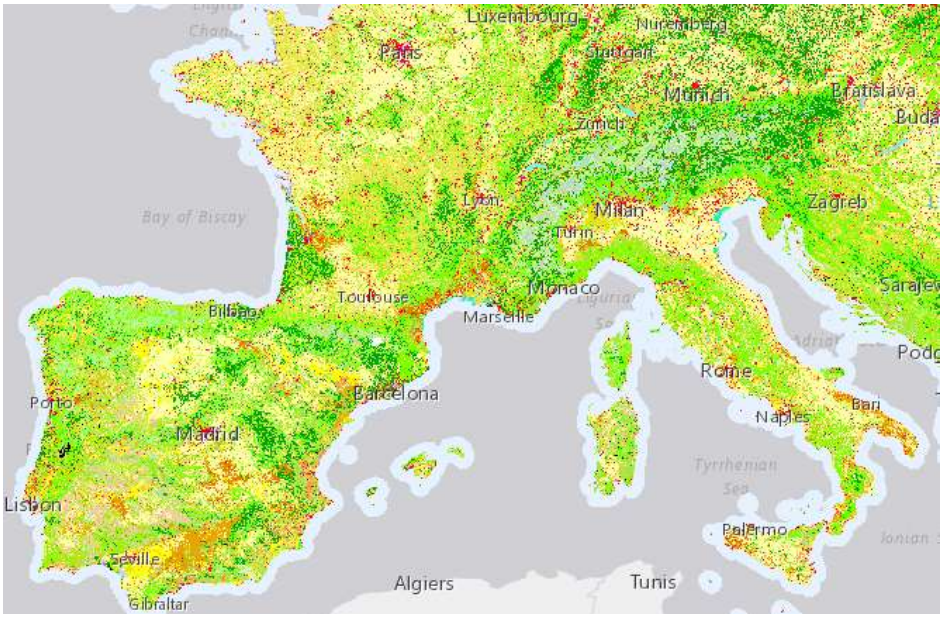


Figure 30 - CORINE 2018 subset (Italy, Spain, and central Europe)

The original raster file reference system in this case was in Lambert (*LAEA* Europe also called *ETR89*-extended) with a resolution of $100x100m$, different from the *UTM32* of our project; moreover, it was also different form *WGS84* often used to avoid the reprojection of values. In this case the reprojection was unavoidable, and so the raster file has been transformed directly into *UTM32*. Such reprojection implies a method for the interpolation of values; here the simple "closest neighbor" approach has been used (the resolution of the original file was high enough to not commit big mistakes during this operation). The new raster maintained the same resolution of the original. The file has then been cropped and limited on the area of study.

The last step before the calculation of the land-use *CDs* is the reclassification of the database. As already explained at the beginning of the paragraph, the *CORINE* database subdivides the territory in

44 classes, but we have only 5 descriptors. The reclassification has been performed following the subdivision reported in Table 5:

ORIGINAL CLASS	ORIGINAL ID	NEW CLASS	RECLASSIFIED ID
Continuous urban fabric	1	URBAN	1
Discontinuous urban fabric	2		
Industrial or commercial units	3		
Road and train network and associated land	4		
Port areas	5		
Airports	6		
Mineral extraction sites	7	SMALLVEG	2
Dump sites	8		
Construction sites	9		
Green urban areas	10		
Sport and leisure facilities	11		
Beaches dunes sands	30		
Bare rocks	31		
Sparsely vegetated areas	32		
Burnt areas	33		
Glaciers and perpetual snow	34		
Non-irrigated arable land	12	AGRI	3
Permanently irrigated land	13		
Rice fields	14		
Vineyards	15		
Fruit trees and berry plantations	16		
Olive groves	17		
Pastures	18		
Annual crops associated with permanent crops	19		
Complex cultivation patterns	20		
Land principally occupied by agriculture with significant areas of natural vegetation	21		
Agro-forestry areas	22		
Broad-leaved forest	23	FOREST	4
Coniferous forest	24		
Mixed forest	25		
Natural grasslands	26		
Moors and heathland	27		
Sclerophyllous vegetation	28		
Transitional woodland-shrub	29		
Inland marshes	35	WETLAND	5
Peat bogs	36		
Salt marshes	37		
Salines	38		

Intertidal flats	39		
Water courses	40		
Water bodies	41		
Coastal lagoons	42		
Estuaries	43		
Sea and ocean	44	NODATA	NA
NODATA	-		

Table 5 - Corine riclassification scheme

For each new class is important to recall the related *CD* description reported in Table 4: for example, inside the smallveg class are included not only the low form of vegetation, but also many tiles of bare soil in the original raster file that barely have anything in common with the vegetation at all (like the permanent glaciers or the bare rocks). It is also important to underline how the sea and ocean tiles (in our area in the south), originally classified as a separate category, in the new reclassification are joint with the tiles where no information was instead already available and not collocated, as one could expect, inside the wetland class. The new reclassified raster file is reported below:

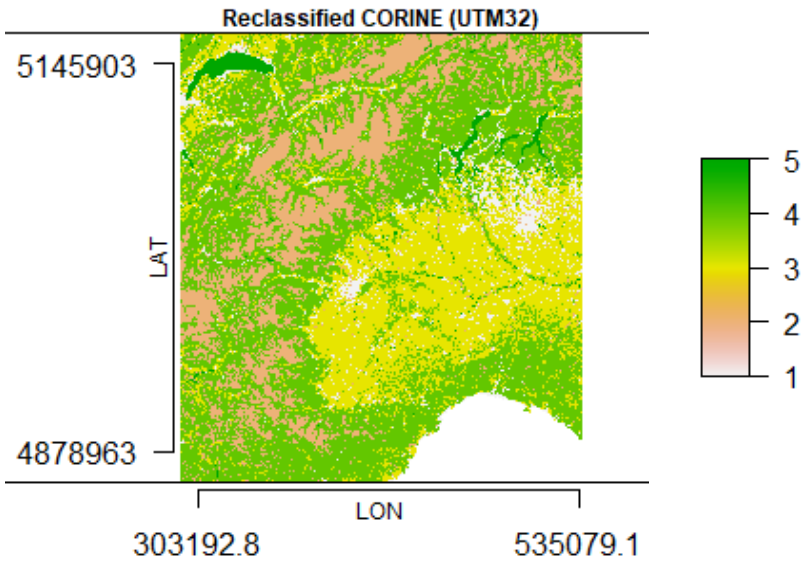


Figure 31 - Reclassified CORINE land use (UTM32)

The five *CDs* are all calculated following the same approach: like for the elevation ones, being the resolution of the raster file higher than the grid in *UTM32* (and so there are more cells of the raster per pixel of the grid), the *"extract"* function has been used to obtain the subsets of raster cells

contained inside each grid pixel. The indicators are expressed as percentage with respect to the total *non-NA* (non-missing value) cells of each subset:

$$LD_{urban_i} = \frac{\sum cell_{1,i}}{\sum cell_i} * 100 \qquad (eq. 5.8a)$$

$$LD_{smallveg_i} = \frac{\sum cell_{2,i}}{\sum cell_i} * 100 \qquad (eq. 5.8b)$$

$$LD_{agri_i} = \frac{\sum cell_{3,i}}{\sum cell_i} * 100 \qquad (eq. 5.8c)$$

$$LD_{forest_i} = \frac{\sum cell_{4,i}}{\sum cell_i} * 100 \qquad (eq. 5.8d)$$

$$LD_{wetland_i} = \frac{\sum cell_{5,i}}{\sum cell_i} * 100 \qquad (eq. 5.8e)$$

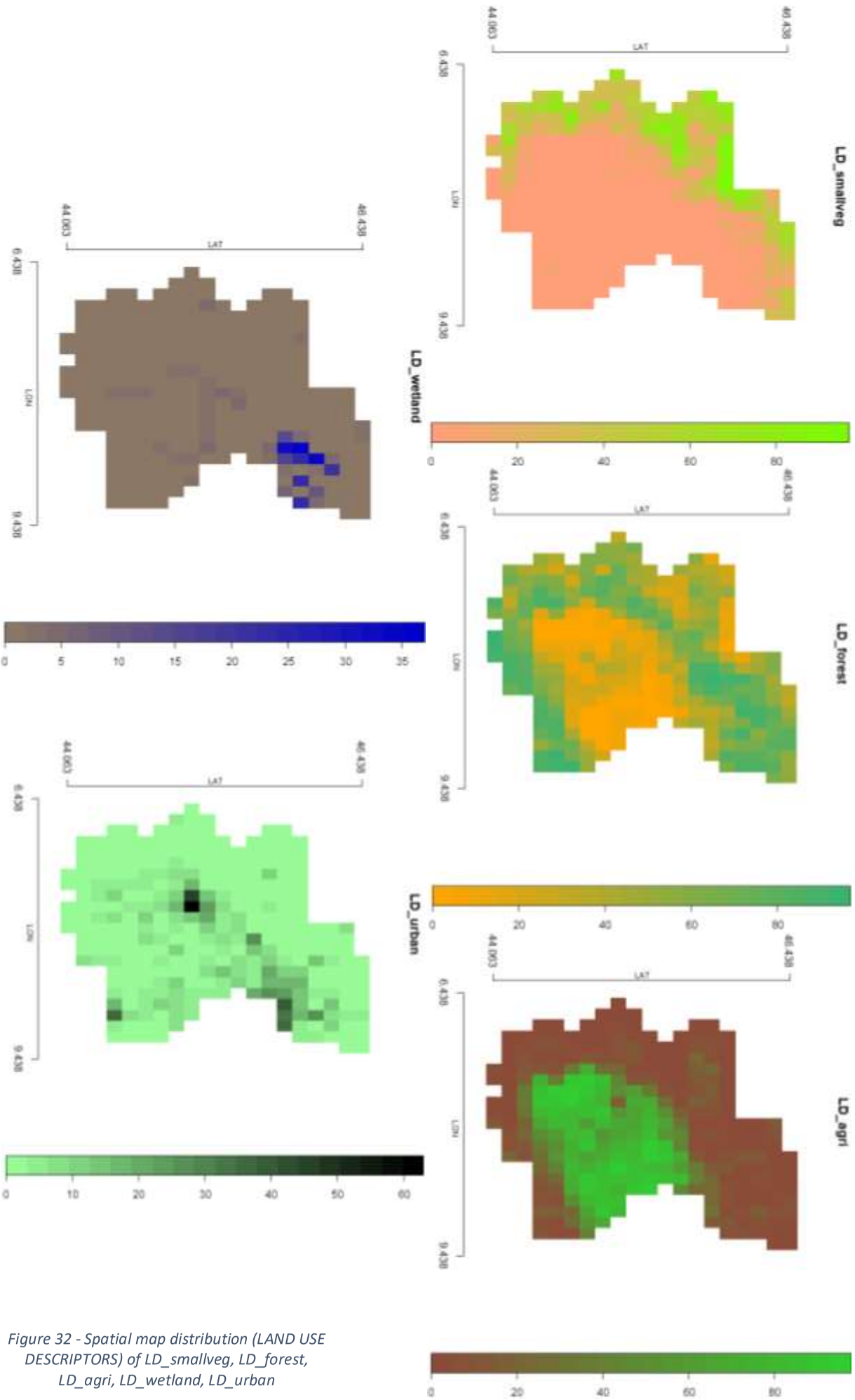


Figure 32 - Spatial map distribution (LAND USE DESCRIPTORS) of LD_smallveg, LD_forest, LD_agri, LD_wetland, LD_urban

5.5 Soil descriptors

"Soil information, from the global to the local scale, has often been the one missing biophysical information layer. The lack of reliable and harmonized soil data has considerably hampered […] environmental impact studies." (Harmonized World Soil Database, FAO, 2009). Such shortage of information has been faced by *FAO* (Food and Agriculture Organization of the United Nations) and *IIASA* (International Institute for Applied Systems Analysis) that combined the vast number of regional and national data together with the already existing digital soil map of the World into the new *Harmonized World Soil Database (HWSD)*. The source data have been retrieved from this database, specifically from the version 1.2. This data has been converted in a series of raster layers, importable in *R*, already with reference system *WGS84* and cropped over the region of interest. The soil *CDs* here considered are most of the descriptors (32 out of 52) and so it will be impractical to report each single original raster layer and final map as for the previous cases. For this reason, the description will be limited to the main calculation strategies and data, while for a complete compendium of the remaining information it is possible to refer to the Attachment 4 (maps and visual information) and Table 5 (*CDs* descriptions). Moreover, in the following formulas the indices from 21 to 52 will be used to indicate each descriptor following the order again of the list in Table 4.

The first eight *CDs* are related to the volumetric percentual abundance ($\%vol$) of particles following the conventional grain size classification, distinguishing the topsoil ($0 - 30\ cm$) from the subsoil ($30 - 100\ cm$):

ISO 14688-1:2002

Name				Size range (mm)	Size range (approx. in)
Very coarse soil		Large boulder	LBo	>630	>24.8031
		Boulder	Bo	200–630	7.8740–24.803
		Cobble	Co	63–200	2.4803–7.8740
Coarse soil	Gravel	Coarse gravel	CGr	20–63	0.78740–2.4803
		Medium gravel	MGr	6.3–20	0.24803–0.78740
		Fine gravel	FGr	2.0–6.3	0.078740–0.24803
	Sand	Coarse sand	CSa	0.63–2.0	0.024803–0.078740
		Medium sand	MSa	0.2–0.63	0.0078740–0.024803
		Fine sand	FSa	0.063–0.2	0.0024803–0.0078740
Fine soil	Silt	Coarse silt	CSi	0.02–0.063	0.00078740–0.0024803
		Medium silt	MSi	0.0063–0.02	0.00024803–0.00078740
		Fine silt	FSi	0.002–0.0063	0.000078740–0.00024803
	Clay		Cl	≤0.002	≤0.000078740

Table 6 - ISO standard 14688-1:2002 for soil classification based on grain size

The original raster layers ware already composed by a percentual indication for each cell; the descriptor have been then simply calculated as the average of the subsets related to each pixel of the reference grid (neglecting as usual missing values):

$$CD_{21to28,i} = \frac{\sum V_{21to28,i}}{N_i} \quad (eq. 5.9)$$

With N_i the number of *non-NA* cells in the i-subset and V_i the values assumed by the cells of the subset.

Next are the reference-bulk densities (also called apparent or volumetric densities) of topsoil and subsoil (kg/dm^3). They are defined as the mass of the soil particles divided by the total volume (particle volume, inter-particle volume and internal pore volume) they occupy. Again, the descriptors are calculated with a simple average of the raster layer cells belonging to the same pixel subset (from now on the formula is not reported anymore).

Then there are three descriptors for three classes of Available Water Content (*AWC*): the available water storage capability is expressed in mm/m and the following classes are used (*Harmonized World Soil Database,* 2009):

CLASS	*AWC* (mm/m)
1	150
2	125
3	100
4	75
5	50
6	25
7	0

Table 7 - AWC classes (HWSD, 2009)

In these cases, the descriptors are about the percentage of pixels having a *LARGE* ($150 - 125\ mm/m$), *MEDIUM* ($75 - 125\ mm/m$) or *SMALL* ($15 - 75\ mm/m$) available water content. The

raster layer associates to each cell the relative class code from Table 7, and the *CDs* have been evaluated as:

$$CD_{31,i} = \frac{\sum C_i\,/(V_i = 1{,}2)}{N_i} * 100 \quad (eq.5.10a)$$

$$CD_{32,i} = \frac{\sum C_i\,/(V_i = 3{,}4)}{N_i} * 100 \quad (eq.5.10b)$$

$$CD_{33,i} = \frac{\sum C_i\,/(V_i = 5{,}6)}{N_i} * 100 \quad (eq.5.10c)$$

With C_i indicating the number of cells per subset meeting the condition in parenthesis. Non considering in any of the three cases class 7, the sum of the three descriptors is not necessarily 100%.

The *IL* (impermeable layer) descriptors indicate the presence or absence of permeable layers along the soil profile expressed as percentages. The layer associates to each cell a number ranging from 0 to 4 following the indications in Table 8:

Code	Impermeable Layer (IL)
0	No information
1	No impermeable within 150 cm
2	Impermeable between 80 and 150 cm
3	Impermeable between 40 and 80 cm
4	Impermeable within 40 cm

Table 8 - Impermeable Layer (IL) classification scheme (HWSD, 2009)

$$CD_{34,i} = \frac{\sum C_i\,/(V_i = 3{,}4)}{N_i} * 100 \quad (eq.5.11a)$$

$$CD_{35,i} = \frac{\sum C_i\,/(V_i = 2)}{N_i} * 100 \quad (eq.5.11b)$$

$$CD_{36,i} = \frac{\sum C_i\,/(V_i = 1)}{N_i} * 100 \quad (eq.5.11c)$$

An analogous approach is used also for the calculation of the *SWR* (Soil Water Regime) related descriptors that indicate the dominant average soil water regime of the soil profile. Also, here a classification has been performed inside the *HWSD*:

Code	Soil Water regime (WR)
0	No information
1	Not wet within 80 cm for over 3 months, nor wet within 40 cm for over 1 month
2	Wet within 80 cm for 3 to 6 months, but not wet within 40 cm for over 1 month
3	Wet within 80 cm over 6 months, but not wet within 40 cm for over 11 month
4	Wet within 40 cm depth for over 11 month

Table 9 - SWR (Soil Water Regime) classification scheme (HWSD, 2009)

$$CD_{37,i} = \frac{\sum C_i\ /(V_i = 1)}{N_i} * 100 \quad (eq.5.12a)$$

$$CD_{38,i} = \frac{\sum C_i\ /(V_i = 2)}{N_i} * 100 \quad (eq.5.12b)$$

$$CD_{39,i} = \frac{\sum C_i\ /(V_i = 3)}{N_i} * 100 \quad (eq.5.12c)$$

$$CD_{40,i} = \frac{\sum C_i\ /(V_i = 4)}{N_i} * 100 \quad (eq.5.12d)$$

The topsoil textural characteristics are summarized by the next three *CDs*. They refer to the simplified textural classes used in the Soil Map of the World (*FAO/UNESCO, 1970-1980*). In the official material describing the database is specified how considering the small scale of the map, just three classes were considered: *coarse textured* (sands, loamy-sands, and sandy-loams with less than 18% clay and more than 65% sand, code 1), *medium textured* (sandy-loams, loams, sandy-clay-loams, silt-loams, silt, silty-clay-loams, and clay-loams with less than 35% clay and less than 65% sand; the sand fraction may be as high as 82% if a minimum of 18% of clay is present, code 2), and *fine textured* (clays, silty-clays, sandy-clays, clay-loams and silty—clay-loams with more than 35% clay, code 3). As for the previous cases the calculation has been performed like:

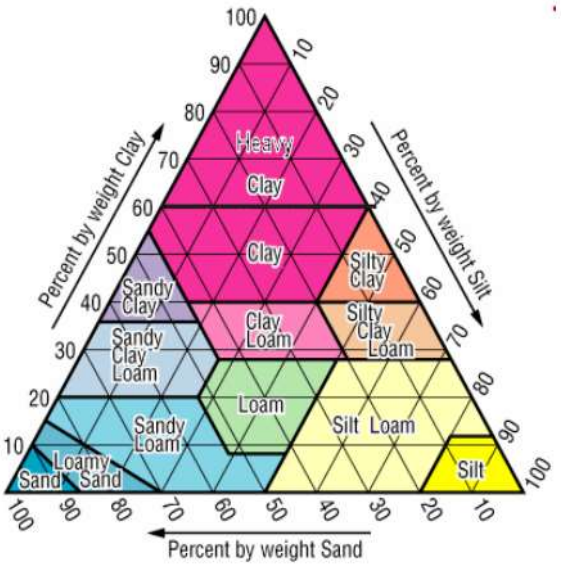
$$CD_{41,i} = \frac{\sum C_i\ /(V_i = 1)}{N_i} * 100 \quad (eq.5.13a)$$

$$CD_{42,i} = \frac{\sum C_i\ /(V_i = 2)}{N_i} * 100 \quad (eq.5.13b)$$

$$CD_{43,i} = \frac{\sum C_i\ /(V_i = 3)}{N_i} * 100 \quad (eq.5.13c)$$

The following six descriptors are the ones related to the *USDA* (U.S. Department of Agriculture) classification. Three are for the topsoil, three for subsoil. Such classification is based on a triangular diagram having at the three vertices the 100% by weight of one between clay (less than $0.002\ mm$),

sand (between 0.002 and $0.05\ mm$) and silt (between 0.05 and $2\ mm$) following again the *ISO-standard 14688-1* about grain size distribution already used previously. Moving along the sides of the triangle (depending on the reciprocal percentages of the components), it is possible to identify a single point inside it, which has associated a unique class:



Code	Texture
1	clay (heavy)
2	silty clay
3	clay
4	silty clay loam
5	clay loam
6	silt
7	silt loam
8	sandy clay
9	loam
10	sandy clay loam
11	sandy loam
12	loamy sand
13	sand

Figure 33 - USDA textural triangle and classes classification (HWSD, 2009)

$$CD_{44,i} = \frac{\sum C_i\ /(V_i = 1,2,3,8)}{N_i} * 100 \quad (eq.5.14a)$$

$$CD_{45,i} = \frac{\sum C_i\ /(V_i = 1,2,3,8)}{N_i} * 100 \quad (eq.5.14b)$$

$$CD_{46,i} = \frac{\sum C_i\ /(V_i = 12,13)}{N_i} * 100 \quad (eq.5.14c)$$

$$CD_{47,i} = \frac{\sum C_i\ /(V_i = 12,13)}{N_i} * 100 \quad (eq.5.14d)$$

$$CD_{48,i} = \frac{\sum C_i\ /(V_i = 4,5,6,7,9,10,11)}{N_i} * 100 \quad (eq.5.14e)$$

$$CD_{49,i} = \frac{\sum C_i\ /(V_i = 4,5,6,7,9,10,11)}{N_i} * 100 \quad (eq.5.14f)$$

The last set of three *CDs* is the one describing the drainage classes as defined in the *FAO* 2006 guidelines. The possible classes are six: *excessive*, *well*, *moderate*, *imperfect*, *poor* and *very poor*. As

usual the related raster layer associates to each cell a value from 6 to 1 to classify them into one of the six:

$$CD_{50,i} = \frac{\sum C_i\,/(V_i = 5{,}6)}{N_i} * 100 \quad (eq.5.15a)$$

$$CD_{51,i} = \frac{\sum C_i\,/(V_i = 3{,}4)}{N_i} * 100 \quad (eq.5.15b)$$

$$CD_{52,i} = \frac{\sum C_i\,/(V_i = 1{,}2)}{N_i} * 100 \quad (eq.5.15c)$$

5.6 Curve numbers

The final group of descriptors utilized in this work are the curve numbers (*CN*). The curve numbers were developed inside the so-called *SCS-CN* method by the *USDA* Natural Resource Conservation Center; such method is utilized to approximately determine the amount of directly generated runoff from a rainfall event over an area. The method can be formulated as follows:

$$Q = \begin{cases} 0 & for\ P \leq I_a \\ \dfrac{(P - I_a)^2}{P - I_a + S} & for\ P > I_a \end{cases} \quad (eq.5.16)$$

Where Q is the generated runoff, P is the rainfall precipitation, I_a is the so-called initial abstraction and S is the maximum potential soil moisture retention after runoff begins. It is generally assumed that the initial abstraction can be calculated as:

$$I_a = 0.2S \quad (eq.5.17)$$

Practically the missing parameter that must be evaluated for the application of the *SCS-CN* method is S. The empirical relation to do so directly involves the *CN*:

$$S = \frac{25400}{CN} - 254 \quad (eq. 5.18)$$

A curve number is an empirical number, that ranges from 30 to 100, from low runoff potential to high runoff potential, that depends on several parameters: in first place on the *soil type*; there are four categories (from class *A* to class *D*) with different soil infiltration capacities:

Infiltration capacity of the soil	Soil type
High (deep sand and gravels)	A
Moderate (moder. deep sandy soils, loamy sand)	B
Low (flat sandy soil, sandy loam)	C
Very low (clay soils)	D

Table 10 - Soil types for the CNC-CN method

The second parameter is the *land use*: the same soil typology, if used for agricultural purposes or covered by an impermeable road, will present completely different infiltration properties. Finally, the *moisture class*: depending on the season and the amount of rainfall fallen in the previous five days, one of three possible outcome is selected; each of these results is identified by a different curve number (*CN1*, *CN2* or *CN3*). In case there are the condition to be in class I or III, the values are determined starting in any case from the value for moisture class II:

Rainfall amount in the 5 previous days		
growing season	non-growing seas.	Soil moisture class
< 30 mm	< 15 mm	I
30 – 50 mm	15 – 30 mm	II
> 50 mm	> 30 mm	III

Table 11 - Determination of the moisture class

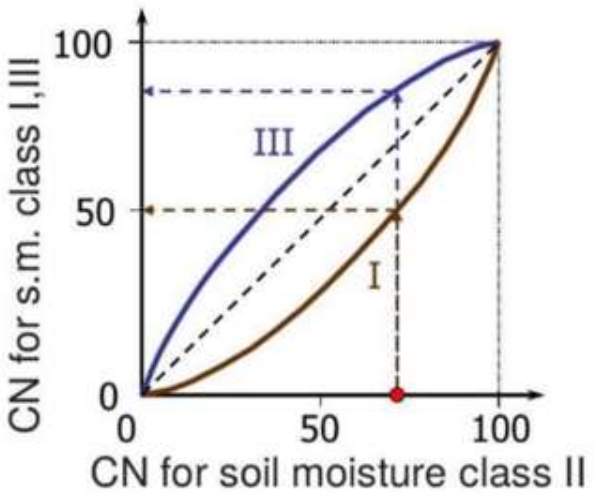


Figure 34 - Conversion diagram from class II to classes I and III

For the first step (the determination of the soil type) a classification has been conducted based on the volumetric percentual composition of clay, sand, and silt (the data are again the one from the *HWSD* already used in the previous paragraph) employing the following sets of conditions:

$$\begin{cases} S_{sand_i} \geq 70 \quad and \quad \left(S_{clay_i} + S_{silt_i}\right) \leq 30 \\ T_{sand_i} \geq 70 \quad and \quad \left(T_{clay_i} + T_{silt_i}\right) \leq 30 \end{cases} \rightarrow \quad Class = A$$

$$\begin{cases} S_{sand_i} \geq 50 \quad and \quad \left(S_{clay_i} + S_{silt_i}\right) \leq 50 \\ T_{sand_i} \geq 50 \quad and \quad \left(T_{clay_i} + T_{silt_i}\right) \leq 50 \end{cases} \rightarrow \quad Class = B$$

$$\begin{cases} S_{sand_i} \geq 20 \quad and \quad \left(S_{clay_i} + S_{silt_i}\right) \leq 80 \\ T_{sand_i} \geq 20 \quad and \quad \left(T_{clay_i} + T_{silt_i}\right) \leq 80 \end{cases} \rightarrow \quad Class = C$$

$$\begin{cases} S_{sand_i} \geq 10 \quad and \quad \left(S_{clay_i} + S_{silt_i}\right) \leq 90 \\ T_{sand_i} \geq 10 \quad and \quad \left(T_{clay_i} + T_{silt_i}\right) \leq 90 \end{cases} \rightarrow \quad Class = D$$

In case of cells that do not satisfy any of these four conditions (generally is the case for missing data about the granulometry shares of the subsoil), the pixel is left empty with a *NA*. The second step is the accounting of the land use; to do so a reference table from *"Carta del Curve Number Regionale" (Agenzia Regionale per la Protezione dell'Ambiente della Sardegna – ARPAS,* 2019*)* has been used. It has been decided to utilize such reference mainly for two reasons: first, it is a document applied yes in a different region of Italy, but still closer to our area then other reference that could be found in literature; second, and probably the most relevant, this table was used in a project relying on similar

data from *CORINE,* like the one used inside this work. So, it has been easy to utilize such reference thanks to a direct correspondence between land use classes of Table 5 with the one reported in Tables 12-13:

Classi di uso del suolo della Corine Land Cover 2008		CN_{II}			
		A	B	C	D
1111	tessuto residenziale compatto e denso	77	85	90	92
1112	tessuto residenziale rado	61	75	83	87
1121	tessuto residenziale rado e nucleiforme	61	75	83	87
1122	fabbricati rurali	59	74	82	86
1211	insediamenti industriali-artigianali e commerciali e spazi annessi	89	92	94	95
1212	insediamenti di grandi impianti di servizi	81	88	91	93
1221	reti stradali e spazi accessori	98	98	98	98
1222	reti ferroviarie e spazi annessi	98	98	98	98
1223	grandi impianti di concentramento e smistamento merci	98	98	98	98
1224	impianti a servizio delle reti di distribuzione	98	98	98	98
123	aree portuali	98	98	98	98
124	aree aereoportuali ed eliporti	98	98	98	98
131	aree estrattive	76	85	89	91
1321	Discariche	76	85	89	91
1322	depositi di rottami a cielo aperto, cimiteri di autoveicoli	76	85	89	91
133	Cantieri	76	85	89	91
141	aree verdi urbane	39	61	74	80
1421	aree ricreative e sportive	49	69	79	84
1422	aree archeologiche	68	79	86	89
143	Cimiteri	68	79	86	89
2111	seminativi in aree non irrigue	61	73	81	84
2112	prati artificiali	68	79	86	89
2121	seminativi semplici e colture orticole a pieno campo	63	73	80	83
2122	Risaie	96	96	96	96
2123	Vivai	68	79	86	89
2124	colture in serra	68	79	86	89
221	Vigneti	66	74	80	82
222	frutteti e frutti minori	62	71	78	81
223	Oliveti	62	71	78	81
231	prati stabili	30	58	71	78
2411	colture temporanee associate all'olivo	62	71	78	81
2412	colture temporanee associate al vigneto	66	74	80	82
2413	colture temporanee associate ad altre colture permanenti	64	73	79	82
242	sistemi colturali e particellari complessi	64	73	79	82
243	aree prevalentemente occupate da colture agrarie con presenza di spazi naturali importanti	64	73	79	82
244	aree agroforestali	64	73	79	82
3111	boschi di latifoglie	36	60	73	79
31121	pioppetti, saliceti, eucalitteti ecc. anche in formazioni miste	36	60	73	79
31122	Sugherete	36	60	73	79
31123	castagneti da frutta	36	60	73	79
31124	altri tipi di latifoglio	36	60	73	79
3121	boschi di conifere	45	66	77	83
3122	arboricoltura con essenze forestali di conifere	45	66	77	83
313	boschi misti di conifere e latifoglie	36	60	73	79
321	aree a pascolo naturale	49	69	79	84
3221	cespuglieti e arbusteti	35	56	70	77
3222	formazioni di ripa non arboree	35	56	70	77
3231	macchia mediterranea	35	56	70	77
3232	Gariga	35	56	70	77
3241	aree a vegetazione boschiva e arbustiva in evoluzione: ricoloniz. natuale	43	65	76	82
3242	aree a vegetazione boschiva e arbustiva in evoluzione: ricoloniz. artificiale	43	65	76	82

Table 12 - Land use classes to CNs depending on soil class – Part 1 (ARPAS, 2019)

Classi di uso del suolo della Corine Land Cover 2008		CN_{II}			
		A	B	C	D
3311	spiagge di ampiezza superiore a 25 m	49	68	79	84
3312	aree dunali non coperte da vegetazione di ampiezza superiore a 25 m	49	68	79	84
3313	aree dunali coperte da vegetazione di ampiezza superiore a 25 m	49	68	79	84
3315	letti di torrenti di ampiezza superiore a 25 m	98	98	98	98
332	pareti rocciose e falesie	76	85	89	91
333	aree con vegetazione rada >5% e <40%	63	77	85	88
411	paludi interne	98	98	98	98
421	paludi salmastre	98	98	98	98
422	Saline	98	98	98	98
423	zone intertidali (zona del litorale che dipende dalle maree)	98	98	98	98
5111	fiumi, torrenti, fossi	98	98	98	98
5112	canali e idrovie	98	98	98	98
5121	bacini naturali	98	98	98	98
5122	bacini artificiali	98	98	98	98
5211	lagune, laghi e stagni costieri a produzione ittica naturale	98	98	98	98
5212	acquacolture in lagune, laghi e stagni costieri	98	98	98	98
5213	estuari e delta	98	98	98	98
5231	aree marine a produzione ittica naturale	98	98	98	98
5232	acquacolture in mare libero	98	98	98	98
523	Mare	98	98	98	98

Table 13 - Land use classes to CNs depending on soil class – Part 2 (ARPAS, 2019)

At this point for each cell has been obtained the *CN2*. The *CN* and *CN3* for the different moisture classes are instead calculated with the following formulas, that analytically execute the transformation in Figure 30:

$$CN = \frac{CN2}{2.3 - 0.013CN2} \quad (eq. 5.19a)$$

$$CN3 = \frac{CN2}{0.43 + 0.0057CN2} \quad (eq. 5.19b)$$

The actual *CDs* over the reference grid has been finally calculate with the usual average over the subsets. In Figure 35 are reported the final maps for the three curve numbers:

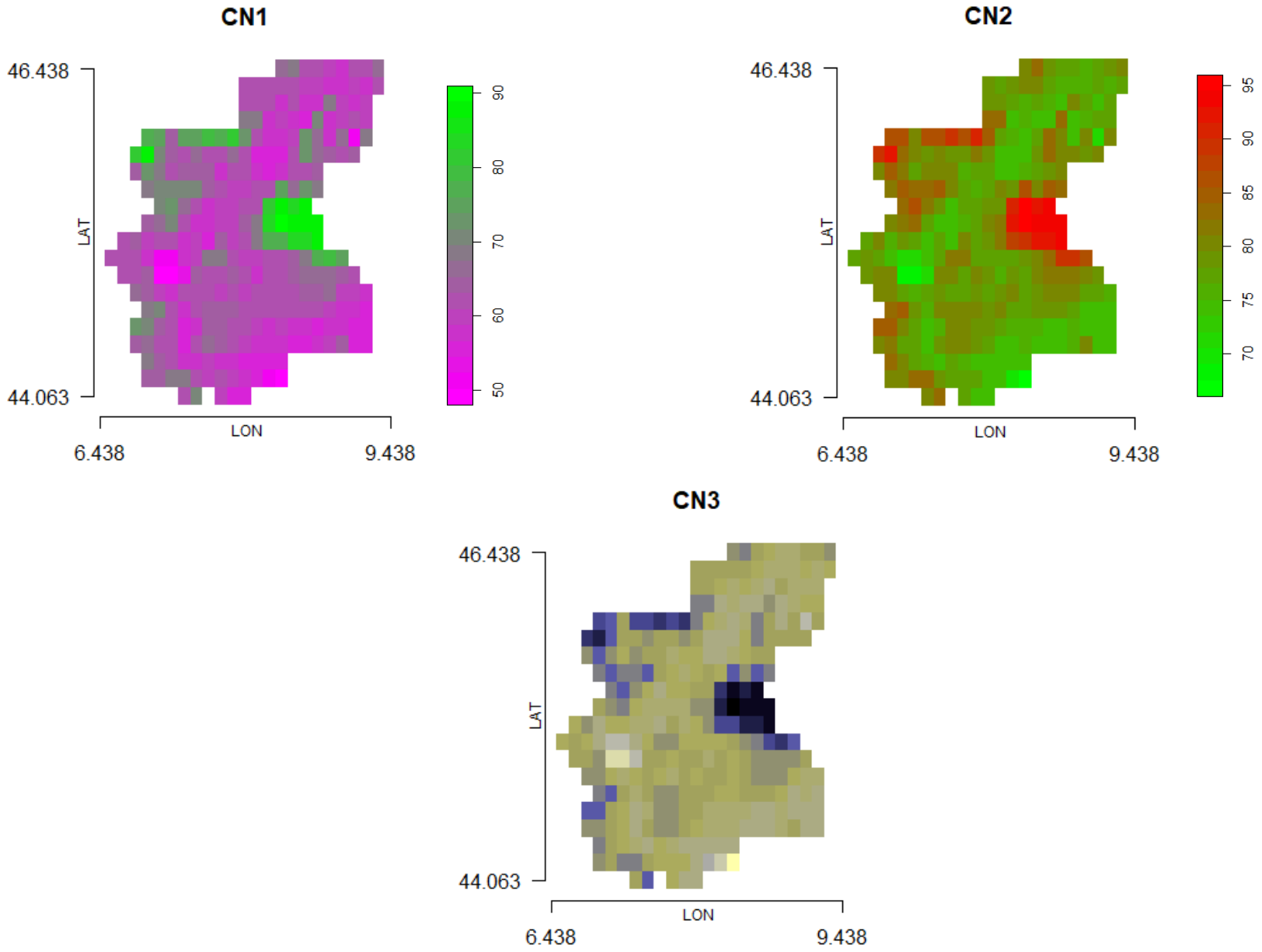
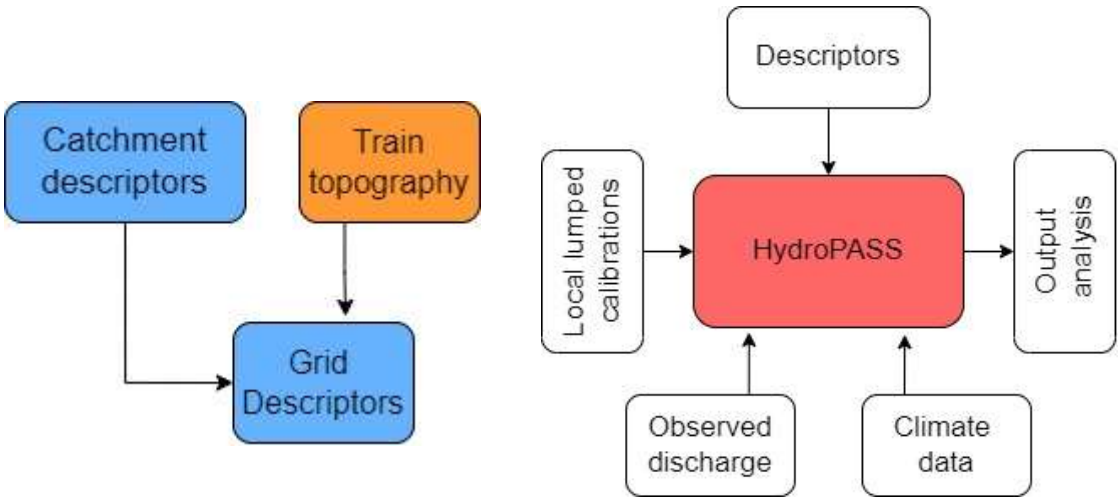


Figure 35 - Spatial map distribution of CN1, CN2, CN3

6. HydroPASS



6.1 Theoretical description

In this chapter will be provided a deeper description of what is the core of the entire work, the *HydroPASS* algorithm. Before entering in the details is a good idea to recap the input data described in the previous three chapters and the main idea behind the regional optimization strategy and to introduce some notations for a better and clear comprehension of the formulas described below.

The main idea behind *HydroPASS* is the determination of a ***REGIONAL FUNCTIONAL RELATIONSHIP*** between catchment descriptors and model parameters: starting from the lumped set of parameters obtained in chapter 4 and the descriptors calculated over the same basins, a relation is found; then it is applied inversely to obtain, starting from the descriptors evaluated over the grid (chapter 5), the distributed model parameter sets over the same pixels. Of all the elements listed above, everything has already been described in detail, with the only exception of the descriptors for the catchment. They are nothing else than the average of the same indicators evaluated over the grid but weighted with the weight of the individual pixels touched by each basin. Such weights have been already used previously and are contained in what was called *"train topology"* and so:

$$CD_{c,i} = CD_{g,j} * W_{j,i} \quad (eq. 6.1)$$

With $CD_{c,i}$ the descriptor for the i-catchment, $CD_{g,j}$ the descriptor over the j-pexel of the grid and $W_{j,i}$ the weight of the i-catchment for the j-pexel. A simpler alternative way could have been to just

consider an average of the *CDs* values of the pixel touched by a catchment, without accounting for the weights; this second approach can be used with small or negligible differences if the resolution of the reference grid is higher. In our case it could have caused some noticeable effect. In Table 14 it is instead reported the notation that will be utilized for the reminder of the chapter:

Descriptor over the grid	$\boldsymbol{CD_g}$
Descriptors over the catchments	CD_c
Train topology/Weights	W
Model parameter sets over the grid	Par_g
Model parameter sets over the catchments	Par_c

Table 14 - Notation

Following the description provided in the original article by *Merz et al.* in 2020, the *HydroPASS* algorithm is composed by 5 iterative steps:

1. *Selection of locally calibrated parameters sets to be used for regionalization*: in this first step, for each catchment, a set of parameters Par_c among the ones considered *"good"* is randomly selected from a pool and then used for the regionalization. The selection of the criteria following which a parameter is considered good or not is not fixed and can be selected by the users. In the original article a set was considered selectable if its $ME > 0.95 * ME_{max,i}$ (or in other words if it belonged to the best 5% of the available parameter for the specific basin). In our case the selection has been made differently: we have considered as good catchments, and not parameters, all the ones which presented a $ME_{max} > 0.8$ and all the 30 sets obtained for such basins as good sets eligible for the random selection. Applying this criterion, of the 117 catchments originally calibrated, only 91 have been kept inside *HydroPASS*.

2. *Regionalize parameters by means of machine learning techniques*: this second step represents the core of the entire algorithm. Starting from the parameter sets randomly drawn in step 1, also using the catchment descriptors CD_c of the same 91 basins and computation intelligence tools, a regional functional relationship is determined. The main point behind this application is that the machine learning tool does not need any a priori assumption about the relation between descriptors and model parameters, differently from some conventional regionalization approaches. In the original article, just like in this work, the so-called *"random*

forest" has been used: it is a variant of three-based models already successfully applied in other field of hydrology in the past, e.g., to predict hydrological signatures (*Addor et al., 2018; Snelder et al., 2009*).

The *"random forest"* is nothing more than a collection of *"trees"* that aggregates all their individual results: a *"tree"* is constructed starting from an original data space (in our case the random Par_c and the CD_c) by recursively splitting it into two branches. Generally, the splitting rule is if a selected CD_c is smaller or larger than a threshold. The splitting criteria (which CD_c and which threshold) is selected by the artificial intelligence in such a way *"to minimize the variability of the model parameters in each of the two generated subgroups"* (*Merz et al. 2020*). Each individual tree is constructed starting from a subsample of the original data space: the subsamples are generated with a *bootstrap sampling*; only around two thirds of CD_c/Par_c data are used for an individual tree, while the other third (also called Out-of-Bag observations) is left out. In R this operation is performed using the *"rpart"* package. Overall, the subset of each tree is partially different from any of the other, resulting each time in a different result. The splitting in branches terminates when a minimum terminal node size is reached. Each forest is composed by 200 trees:

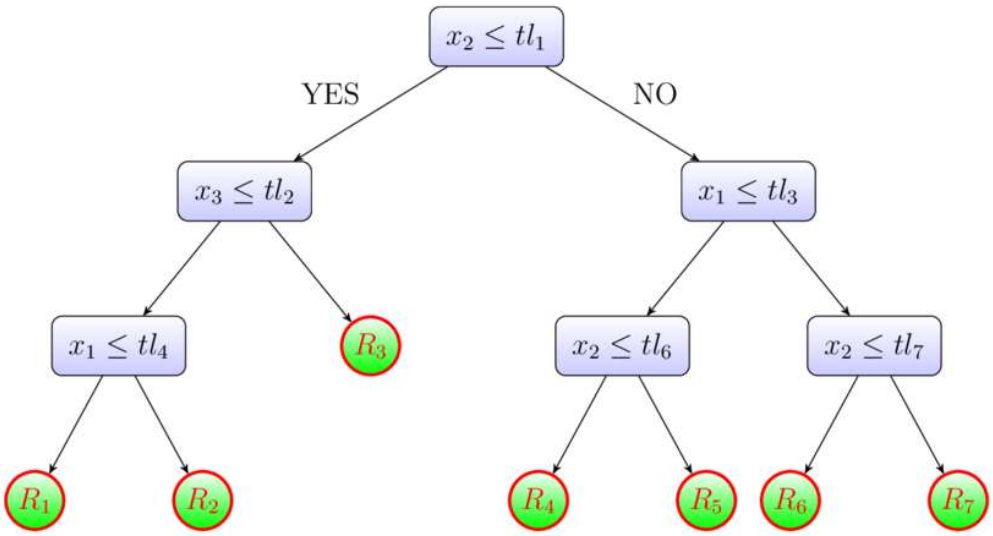


Figure 36 - Schematic of a decision tree (Drivers of economic and financial integration: A machine learning approach, Amir Akbari et al., 2020)

At the end of step 2 is obtained, through the *"random forest"*, what will be from now on called temporary regional functional relationship (RFR_{temp}). It is defined as temporary because it is

not necessarily, and probably will not be, being only at the first iteration, the definitive relationship at the end of the *PASS* algorithm:

$$RFR_{temp} = \begin{cases} Par_{c,1} = f_1(CD_{c,1}, \dots, CD_{c,52}) \\ \dots \\ \dots \\ Par_{c,15} = f_{15}(CD_{c,1}, \dots, CD_{c,52}) \end{cases} \quad (eq.\,6.2)$$

3. <u>*Improve identification of satisfactory regional functional relationships*</u>: this third step is considered optional but proven useful to speed up the determination of better RFR. Once the temporary regional functional relationship RFR_{temp} has been determined at the end of step 2, it is applied to calculate, starting from the CD_c, the so-called *"predicted lumped model parameters"* $Par_{cp,n}$ for each catchment: in simple terms, these new parameters compose a new set for each catchment following the regionalization rule. After that, for each basin, a new parameters' set from the pool of *"good"* set is again selected but this time not randomly: the most similar set to the predicted one is extracted. Such assessment is done evaluating the distance, for each set of each catchment, between the original parameters vector and the predicted parameter vector for the relative catchment. The most similar is the one with the smaller Euclidean distance:

$$dist_i = \sum_{n=1}^{15} (Par_{c,n} - Par_{cp,n})^2 \quad (eq.\,6.3)$$

The new selection of i parameter sets is then reintroduced in the machine learning "random forest" routine. A new RFR_{temp} is determined and step 3 can be repeated. The number of times this iterative step is conducted can be freely selected.

4/5. <u>*Predict regionally consistent parameters for the lumped and distributed model/run the distributed model using regional parameter sets*</u>: the RFR_{temp} obtained after steps 2 and 3 is then applied to calculate the predicted lumped and distributed (over the grid) parameter sets, Par_{cp} and Par_{gp}, using CD_c and CD_g respectively . With the predicted lumped sets of the catchments two operations are performed: first, the local *ME* is evaluated for each basin; this is done in the same way as in chapter 4, using *TUWmodel* in a lumped way and with objective

function the Kling-Gupta efficiency. To do so inside the *HydroPASS* algorithm also the observed climate data over the reference grid for the usual 60-years and the observed discharges for all the utilized catchments must be provided. The average $\overline{ME}_{lumped}$ is the mean of all the obtained model efficiencies. The second operation is an update of the parameters set pool for the extraction in step 1: *"The PASS method test which combination of good local parameter sets gives the best regional prediction. Hence, the approach strongly depends on the availability for each catchment of a sufficient number of good, lumped parameter sets" (Merz et al., 2020).* To update and increase the available number of sets for each basin, the following check is performed: if the predicted lumped set for a catchment well performs (i.e., it has a good, lumped *ME*), it is added to pool. The assessment of what good means in this case follows the conditions:

$$\begin{cases} ME_{p,i} > 0.95 * ME_{max,i} \\ R^2 < 0.95 \ \forall \ already\ available\ sets\ of\ catchment\ i \end{cases} \quad (eq.6.4)$$

The first condition is an actual performance limit; the second one is to avoid that two sets too similar are introduced in the selection pool. Otherwise, during the next iterations at step 1, with increasing probability each time, two nearly identical parameter sets could be extracted, limiting the efficiency of the *"random forest"* and so of the RFR_{temp} determination.

With the distributed sets over the grid a similar procedure is conducted, with a simple difference. Distributed parameter sets means that now we have a vector with the 15 *TUWmodel* parameters for each pixel of the reference grid. Each pixel can also be considered as a *HU* (hydrological unit). *TUWmodel* must now be run in a distributed way: it means evaluate the generated discharge over a catchment by summing all the generated discharge contribution of each *HU* touching the catchment itself, opportunely weighted. Inside R the same *"TUWmodel"* package is used, with the only difference that the input parameter sets are no more a single vector, but a matrix with i-columns (parameter) and j-rows (pixel over the basin):

$$Q_{sim} = \sum_{m=1}^{j} Q_{sim,HU,m} \quad (eq.6.5)$$

The simulated discharge is then compared with the observed discharge for the catchment considered through the same objective function and a distribute model efficiency (ME_{reg}) is determined. The average distributed efficiency ($\overline{ME}_{reg}$) is again simply the average of all the basin distributed efficiencies.

6. *Repeating steps 1-5 to improve* $\overline{ME}_{reg}$: the previous steps are repeated several times to improve the regional distributed efficiency, searching for better RFR_{temp} and improving observation reconstruction. The iterations following the first one can rely on the updated parameters' sets pool for selection from step 3). After the last iteration of the algorithm, the temporary regional functional relationship RFR_{temp} will be considered the definitive one (RFR) representing the final regionalization model.

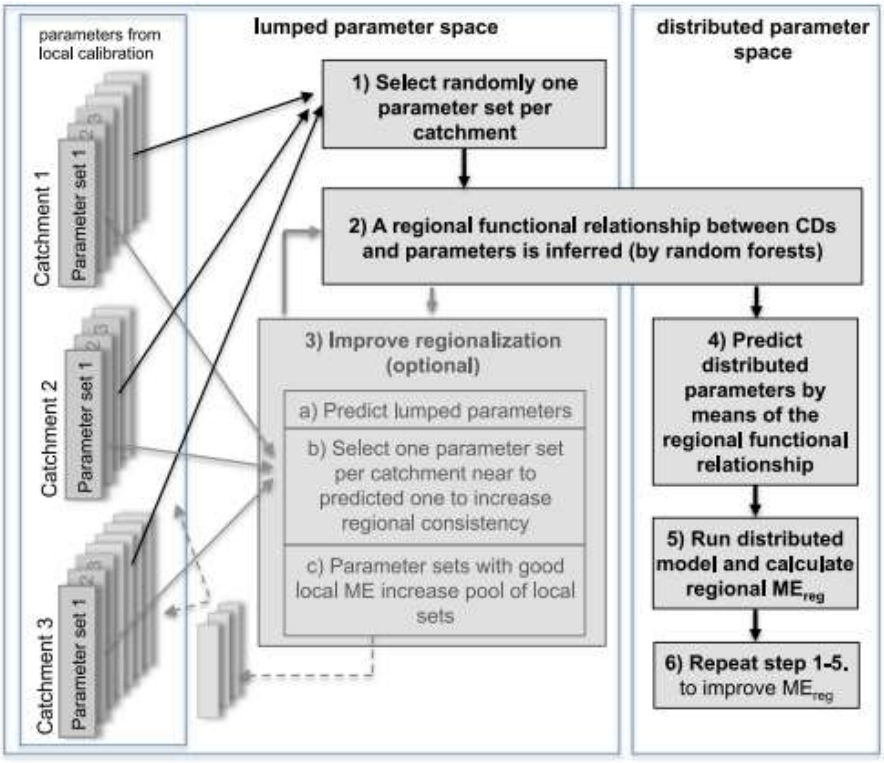


Figure 37 - Flowchart of the Parameter Set Shuffling (PASS) method to derive regionally consistent parameters for distributed models (Merz et al.,2020)

6.2 Practical implementation and options

Inside *R* the algorithm has been implemented with a specific function, appositely programmed. Notice how the same function is now available through the *R* package *"HydroPASS"*, recently made available and described inside the publication *"HydroPASS: a newly developed R package to go through the regional calibration od distributed catchment models" (Pesce et al., 2022)*; nevertheless, the work here proposed has been developed before the publication, when the package was still under development. This will also be the only case in which a closer look to the *R* functions and their structure will be explicitly described: The structure of the function is the following:

$$PASS < -function(Y, X.cat, X.grd, grd2cat, model.eff.fn, lower, upper, PASS.options)$$

The elements are the one already described above, just with different names: Y is a list of a dataframe containing the 30 lumped parameter sets for each catchment or the updated set pool from a previous *PASS* run, $X.cat$ is a matrix or a dataframe of descriptors for each basin, $X.grd$ is also a matrix or a dataframe of descriptors, but in this case over the grid, $grd2cat$ is the *"train topology"*, $model.eff.me$ is the objective function for the *ME* assessment both for the lumped and distributed model, $upper$ and $lower$ are vectors of the extremes boundary for each of the *TUWmodel* parameters (see Table 2) and last $PASS.options$ is a vector containing the setting for the algorithm. Such options are managed by a second distinct function and are the following:

$$PASS.options = function(maxLoops = 100, nGroups = 10, REGloops = 5,$$
$$proportion.max.eff.update = 0.95, sampling = 'random', optim.subset.cat = 0.7)$$

$maxLoops$ determine the maximum number of iterations (i.e., the times the 5 steps are executed) for each individual *PASS* run, $nGroups$, as the name suggests, specifies the number of groups for the regional optimization. This is the first difference with the general scheme proposed by Merz et al.: instead of running the algorithm stand alone, a higher number of groups is used and managed individually or interacting the one with the others (see the reminder of the paragraph). $nGroups$ specify the number of times the improved identification of RFR is performed in step 3, $proportion.max.eff.update$ set the condition to add predicted lumped parameter sets to the draw pool ($eq.6.4$), $sampling$ decide between two possible running modes *("random"* or *"optim"*) and last

$optim.subset.cat$ fix a parameter used inside the second running mode. All the option here indicated are default ones; to change them to different values, they must be specified inside the $PASS$ function.

At the beginning of the algorithm run, n different groups are set (in our case 10 as from default option); each group has associated a *null* starting model efficiency. Depending on the sampling mode two different strategies are applied: if the *"random "* sampling is selected, what described in the general scheme happens; the sets are randomly selected at each iteration from a constantly updated pool of possible parameter sets, a RFR is determined and a $\overline{ME}_{reg}$ efficiency is calculated. This $\overline{ME}_{reg}$, together with all the sets, parameters and *ME* used to obtain it, are saved in the output of the group with the currently lower *ME*, prioritizing the *null* ones. This implies that the first 10 of the 100 iterations of a run just substitute the *null ME* with a valid value; from the 11th iteration moving forward, the worst solution found till that point is substituted by the new one. In case the new solution does not improve any of the existing 10 already saved, it is simply discarded (the only effect transferred to the following iterations is always the updating of the pool of parameter sets).

In case the *"optim"* sampling is chosen instead, there are some differences. First, this second mode cannot be applied if all the 10 groups do not already have a valid value, requiring at least a previous run at "random" to be performed previously. At the beginning a random group is decided. The random selection during step 1 is not completely random: the parameter sets used to obtain the previous solution of the same group are partially substituted to obtain a new extraction partially correlated with the previous one. The number of substituted sets is calculated using the $optim.subset.cat$ option parameter:

$$N_{sub} = N_{cat} * optim.subset.cat \quad (eq.6.6)$$

In our case, maintaining the default option, of the 91 set, 64 are replaced. The sets to be substituted are selected randomly as randomly are selected the substitutive sets from the available pool. After that the algorithm proceeds normally evaluating the $\overline{ME}_{reg}$. In the *"optim"* case the confrontation is always only with the previous *ME* of the same group, and it is not compared with the others. In case it is an improvement, the new information is stored. This second strategy is generally used after the *"random"* one to optimize in a targeted ways single groups and previous solutions after a first general optimization, totally random, has been performed.

The output of the regionalization saved in each group have the following structure:

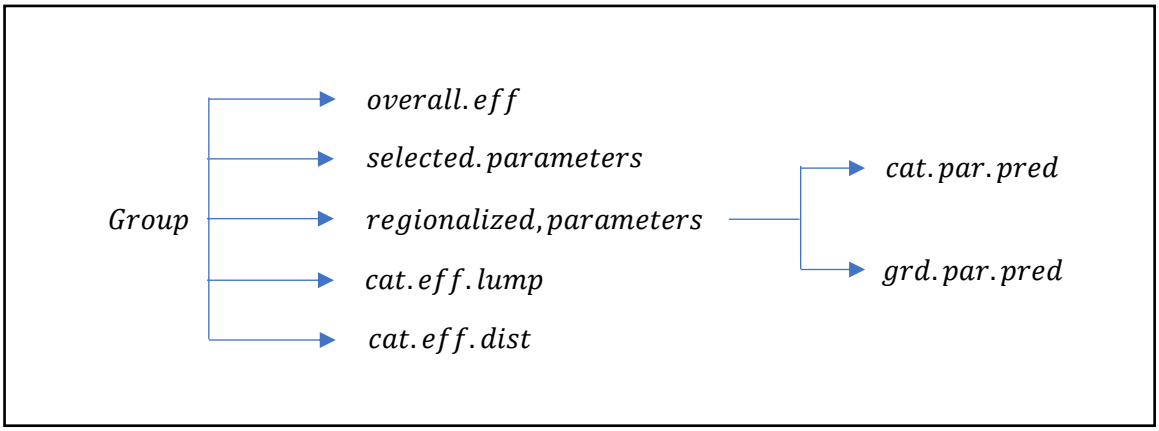


Figure 38 – HydroPASS output data structure

Where $overall.eff$ is the mean of the distributed efficiencies obtained with the RFR, se$lected.parameters$ is a matrix containing the sample parameter sets randomly extracted from the pool, and $cat.eff.dist$ are two vectors containing all the *ME* obtained with the predicted catchment and distributed parameters. Inside $regionalized,parameters$ (two matrices) are instead collected the sets of predicted parameters for each catchment ($cat.par.pred$) and the distributed parameter sets over the reference grid $grd,par.pred$. This last matrix will be of fundamental importance in the next chapter for the application of the regionalization in the prediction of floods over ungauged locations. Additionally, another list ("$train.parameters.updated$") stores the final updated parameter sets pool after the last iteration (and so containing all the possible useful sets).

6.3 Adopted options and results

The results reported in this paragraph have been obtained with the default *PASS* options; just a note about the sampling mode: a first 10 runs (1000 iterations) have been performed in *"random"* mode followed by an additional 10 runs (other 1000 iterations) in *"optim"* mode. The results are reported in Table 15:

```
lumped regionalization efficiencies:
        Group1 Group2 Group3 Group4 Group5 Group6 Group7 Group8 Group9 Group10
Min.     0.538  0.460 -0.030  0.419  0.452  0.401  0.432  0.396  0.471   0.464
1st Qu.  0.691  0.667  0.665  0.654  0.675  0.669  0.672  0.649  0.680   0.676
Median   0.754  0.762  0.750  0.756  0.774  0.756  0.769  0.754  0.766   0.767
Mean     0.742  0.745  0.729  0.729  0.748  0.734  0.744  0.737  0.742   0.745
3rd Qu.  0.821  0.821  0.825  0.811  0.829  0.813  0.825  0.819  0.818   0.829
Max.     0.903  0.901  0.894  0.890  0.910  0.898  0.897  0.906  0.896   0.907
distributed regionalization efficiencies:
        Group1 Group2 Group3 Group4 Group5 Group6 Group7 Group8 Group9 Group10
Min.     0.003  0.203  0.151  0.170  0.090  0.002  0.041  0.048  0.043   0.084
1st Qu.  0.638  0.637  0.656  0.644  0.639  0.649  0.663  0.632  0.632   0.639
Median   0.731  0.721  0.730  0.723  0.727  0.743  0.736  0.739  0.718   0.731
Mean     0.702  0.701  0.703  0.700  0.704  0.701  0.703  0.700  0.699   0.703
3rd Qu.  0.793  0.790  0.809  0.803  0.800  0.794  0.800  0.797  0.804   0.802
Max.     0.901  0.893  0.900  0.883  0.889  0.898  0.897  0.893  0.878   0.895
 updated number of train parameter sets: 47013
```

Table 15 - HydroPASS results

It can be seen, looking the distributed efficiencies, how for all 10 groups the resulting $\overline{ME}_{reg}$ is similar, ranging from 0.699 to 0.704. On the other hand, it is also evident, looking at the minimum and maximum values, the great spread of the results around the mean: in all cases some of the catchments shown a very poor distributed efficiency, while the best ones remained around or below a value of 0.9. If it is true that a lower accuracy was expected after regionalization, it is also true that even the worst performing catchments possessed at least one parameter set whit $ME > 0.8$ after the original local lumped calibration, implying a complete mismatch between the local and regional modelling. Such a relevant decrease in the performances may be an indicator of the incapacity of the distributed model to correctly reconstruct the observed data in some basins. Looking instead at the predicted lumped regionalization efficiencies obtained using the RFR with the CD_c, the average is slightly better, and the spread is considerably more contained, without extremely low (but indeed low anyway) model efficiencies. A global view of the distribution of the model efficiencies for the best regionalization (group 5) is provided below with the confrontation between the *eCDF* of the original lumped efficiencies and the new predicted ones for the catchments (lumped regionalized efficiencies) obtained with *PASS* for the 91 considered basins:

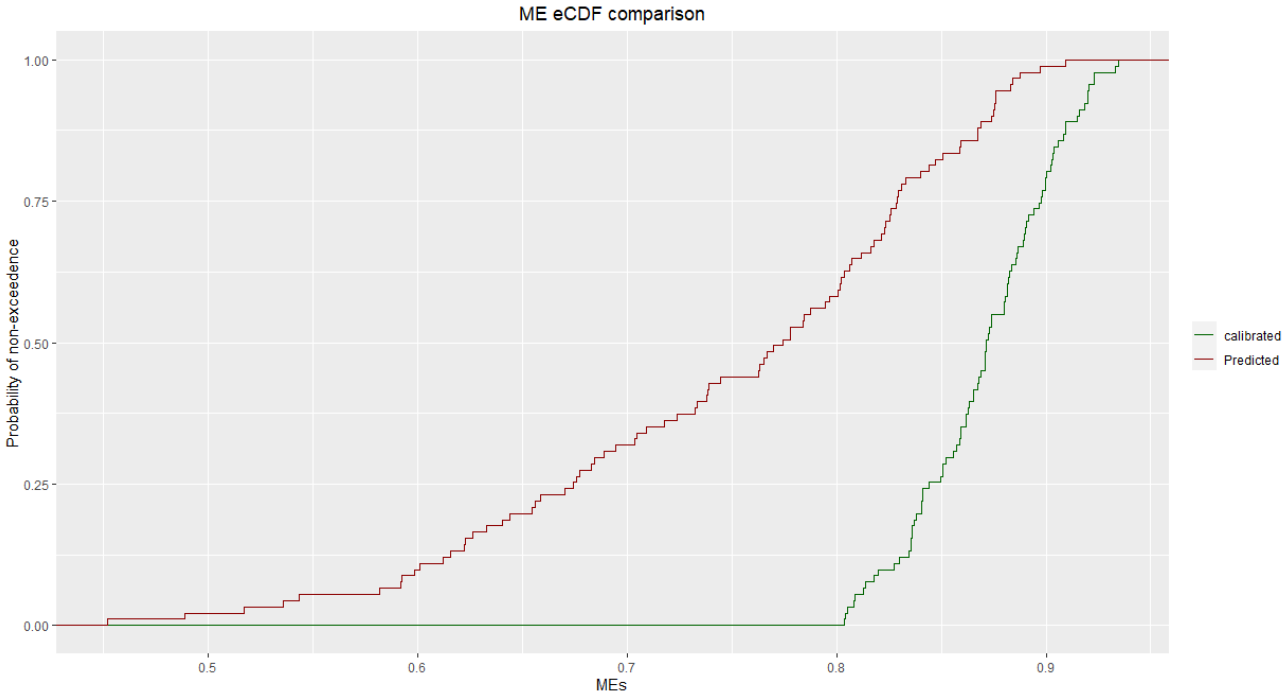


Figure 39 - eCDF for the lumped calibrated ME and predicted local ME for the 91 catchments used inside HydroPASS

We can see, as expected, how after regionalization the distribution shifted toward worst performances as shown by the median and the 25 and 75 percentiles. Nevertheless, despite some catchments that had a lumped local calibration with high performance now present an extremely low *ME*, overall, the performances for most of the basins are still good, with efficiencies greater than 0.6 for more than half the catchments.

Still talking about *MEs,* another aspect has been investigated: Merz et al., in their findings, reached the conclusion that is impossible to establish if a descriptor group (among the 5 listed in chapter 5) is essential for a good regionalization. Such hypothesis has been tested by running again the regionalization with *PASS* by leaving out of the input CD_c and CD_g one by one an entire group of descriptors (for a total of 5 additional regionalizations). The same analysis has been here reproduced: each additional *PASS* regionalization has been done by using the same default options and doing 10 runs (1000 iterations) in *"random"* mode. The choice to not use the 1000 additional iteration in *"optim"* mode is due to computational time requirements; consider in fact that every 100 iterations required between $1.3h$ and the $2.5h$.

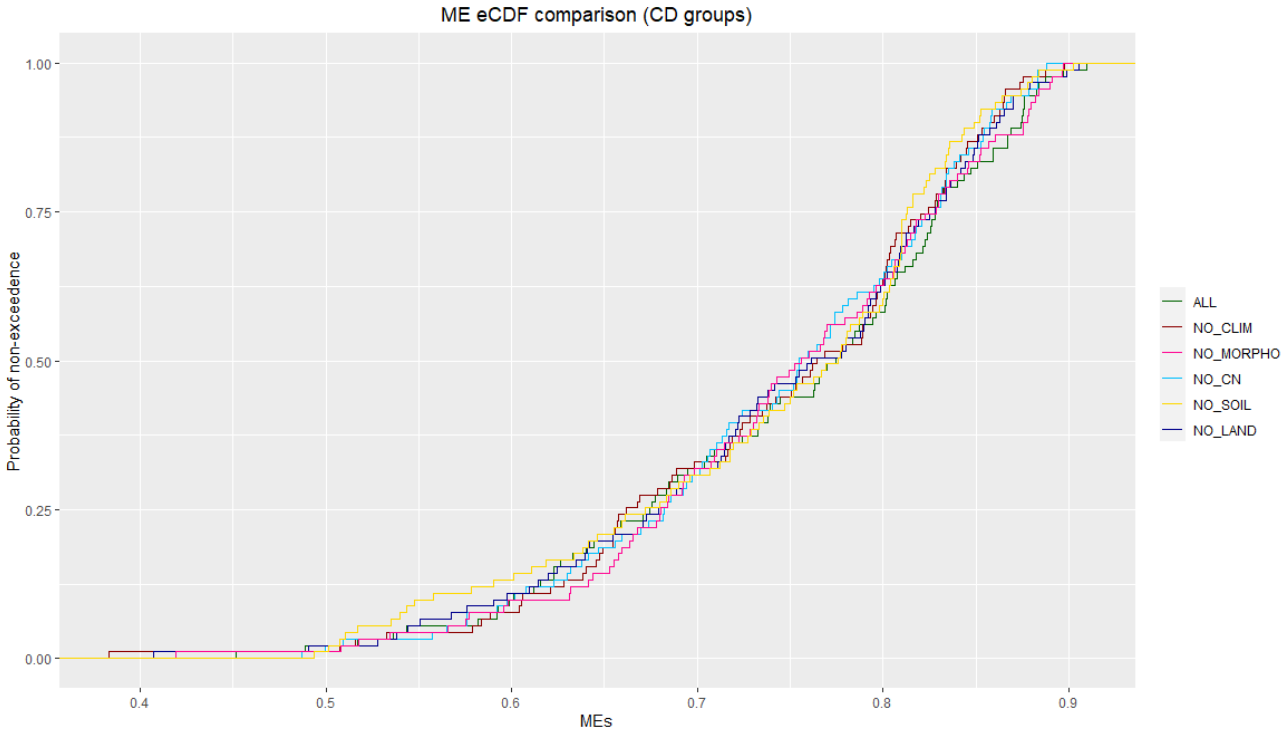


Figure 40 - eCDF comparison (predicted local) for different CDs subsets

The results obtained for the Piedmont region agree with the outcomes reached for the German basins. Removing individual groups of catchment descriptors seemingly does not, or just marginally, influence the *HydroPASS* algorithm performances. This result has positive and negative consequences: on one hand it evidences how the current knowledge of the physical mechanisms behind runoff generation are still uncertain and as consequence is also uncertain the selection of critical descriptors that could be better related the hydrological processes for the definition of an optimal regional functional relation. The result of the comparison can be in fact only explained admitting that each descriptor class is not independent, but instead at least partially correlated, with some of the others. A simple example can be the climatological and the morphological descriptors: climate and morphology developed one in function of the other during geological timescales, embedding and sharing partially the same information. Another case are the *CNs*: they are defined incorporating information about soil classes and land uses, obviously shared with the analogous *CDs* classes. On the positive side there are instead the practical implications: if it is true that using something just because "it works" can be seen as a compromise, on the other hand such tools allow us to face problems always more complex. The awareness of the possibility to remove descriptors imply that also in conditions where such CDs cannot be practically calculated (missing of crucial information or databases) the algorithm can well perform anyway. A lower data requirement makes *HydroPASS* a powerful tool for the regionalization and application of distributed models also in hard conditions or locations with a low monitoring

network diffusion. Moreover, reducing the number of catchment descriptors can significantly decrease the computational time of the simulations.

An additional final analysis that can be performed on the obtained output results is, similarly to what has been already done for the lumped calibrated parameter sets in chapter 4, an investigation about the equifinality. Together with the output data for every group, the predicted lumped parameters iteratively added to the draw pool for each basin are also saved. For each catchment the 10 best available sets (opportunely normalized, see $eq.\,4.15$) among them are analyzed and compared with the best 10 original local lumped calibrated sets. Like already done in chapter 4, only a reference example for the VARPO catchment is reported in Figure 41; a full compendium is available in Attachment 5:

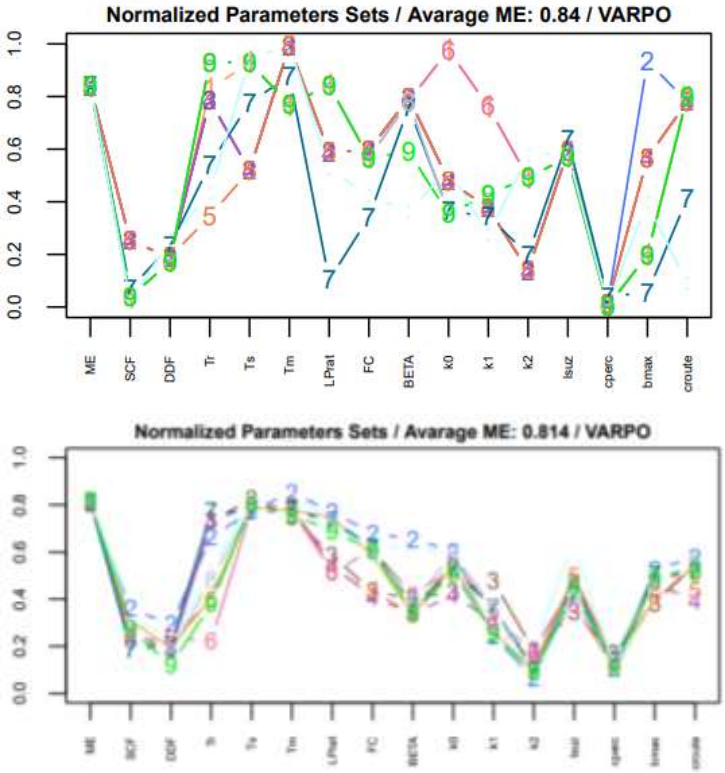


Figure 41 - Best original normalized parameter sets (above) compared with the best 10 normalized predicted parameter sets (below)

The predicted sets found by *PASS* have evidently a strongly reduced equifinality for many parameters if compared with the original ones, with a reduced spread and a higher concentration around average values. This result is interesting because it suggests that some sort of stronger relation between specific parameter values and *ME* is indeed present and *PASS* was able to find and exploit it. The reduced uncertainty indicates that some sort of physical relation can exist and that despite wide range of parameters can be able to reconstruct observed discharge, some correlation between better ME and a smaller range is indeed possible. This aspect is not further investigated in this thesis but could be of great interest for a better understanding of the physical mechanisms behind hydrological processes. This behavior is general for all the 91 considered catchments.

The second graph (Figure 42) shows instead the comparison of the parameters' distributions with boxplots between the original and the predicted sets:

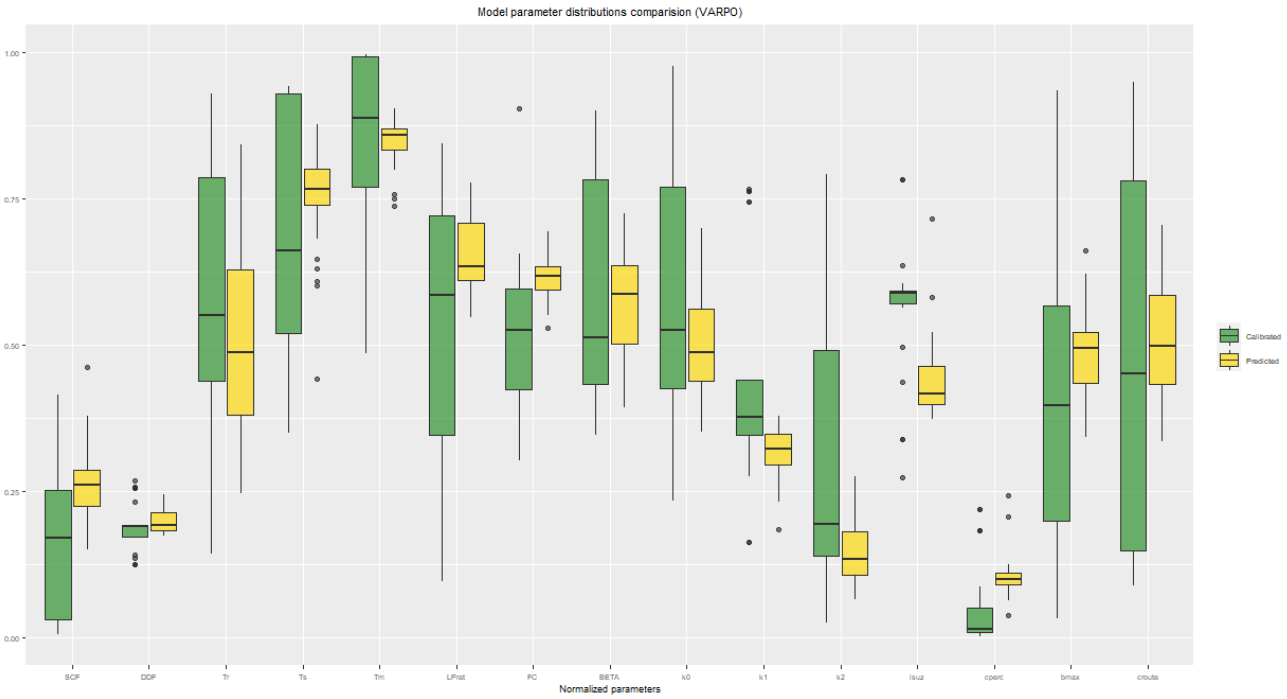


Figure 42 - Model parameter comparison between calibrated and predictor sets (VARPO)

From this comparison it can be better seen how few parameters, generally among the ones that presented a very low equifinality during the lumped local calibration, are in counter tendency with the others, showing an increase of spread around the central value. Few other parameters kept instead a similar uncertainty. This behavior is again general for most of the catchments. As usual, a complete collection for the other basins is available in Attachment 6.

As already stated in paragraph 6.1, the most important outputs of the regionalization are, in the end, the distributed parameter sets over the grid. This parameter sets can be used to run *TUWmodel* in a distributed way to simulate discharge over ungauged areas by summing the runoff generated upon every hydrological unit (*HU*) of interest. In the Figures from 43 to 46 are reported the maps of all 15 distributed parameters as obtained in group 5 with the *HydroPASS* regionalization:

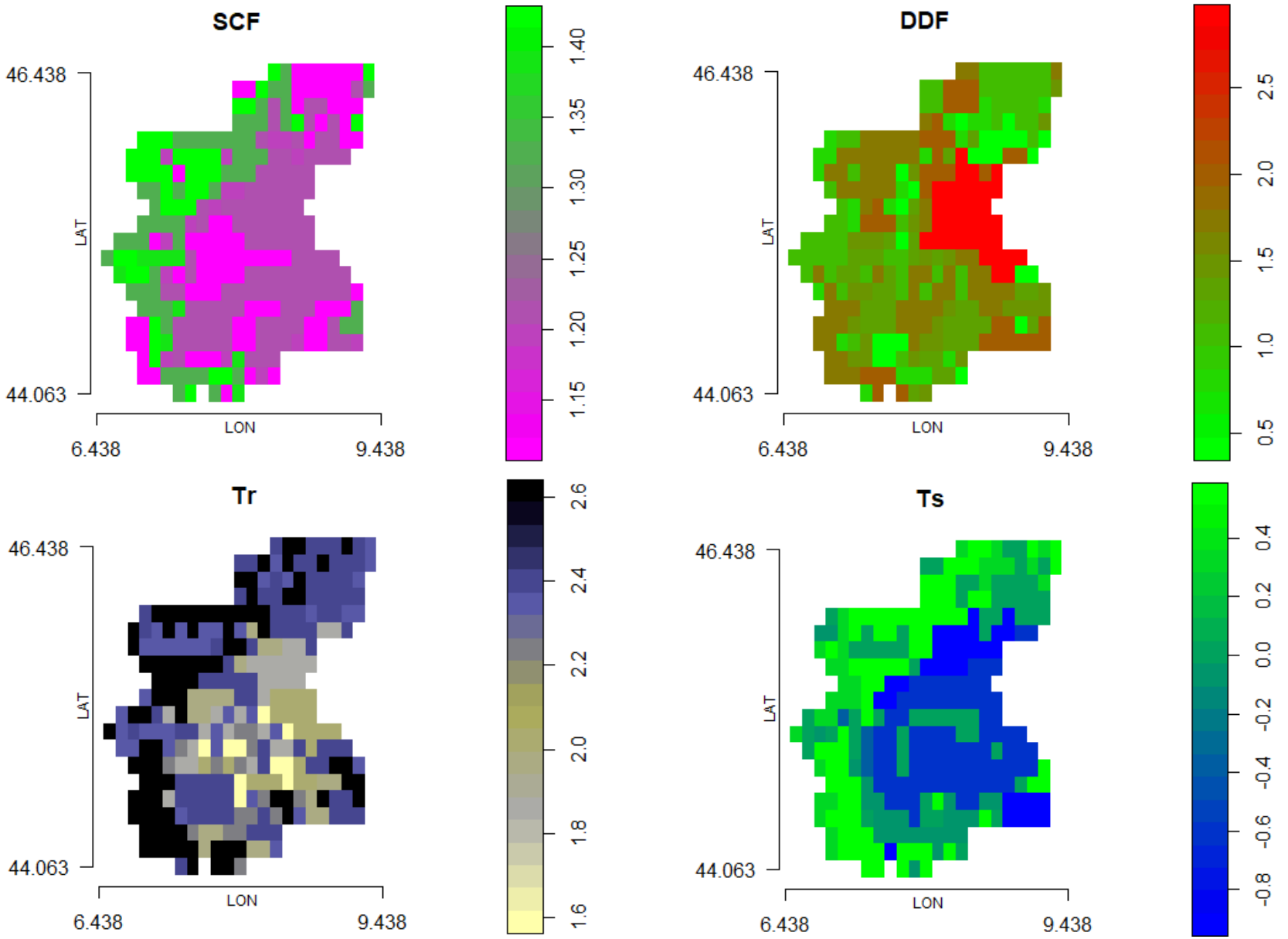


Figure 43 - Spatial map distribution (MODEL PARAMETERS) of SCF, DDF, Tr, Ts

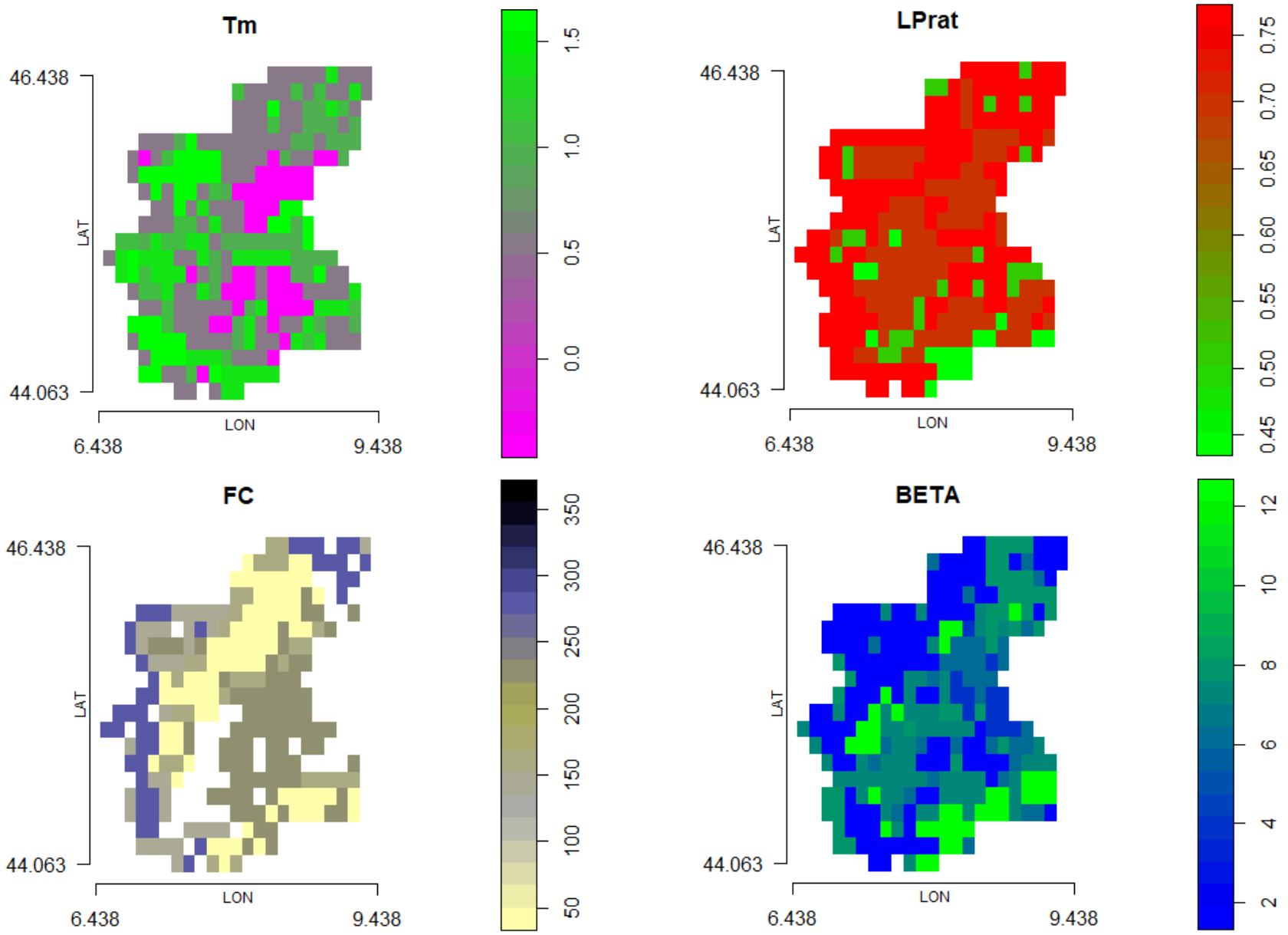


Figure 44 - Spatial map distribution (MODEL PARAMETERS) of Tm, LPrat, FC, BETA

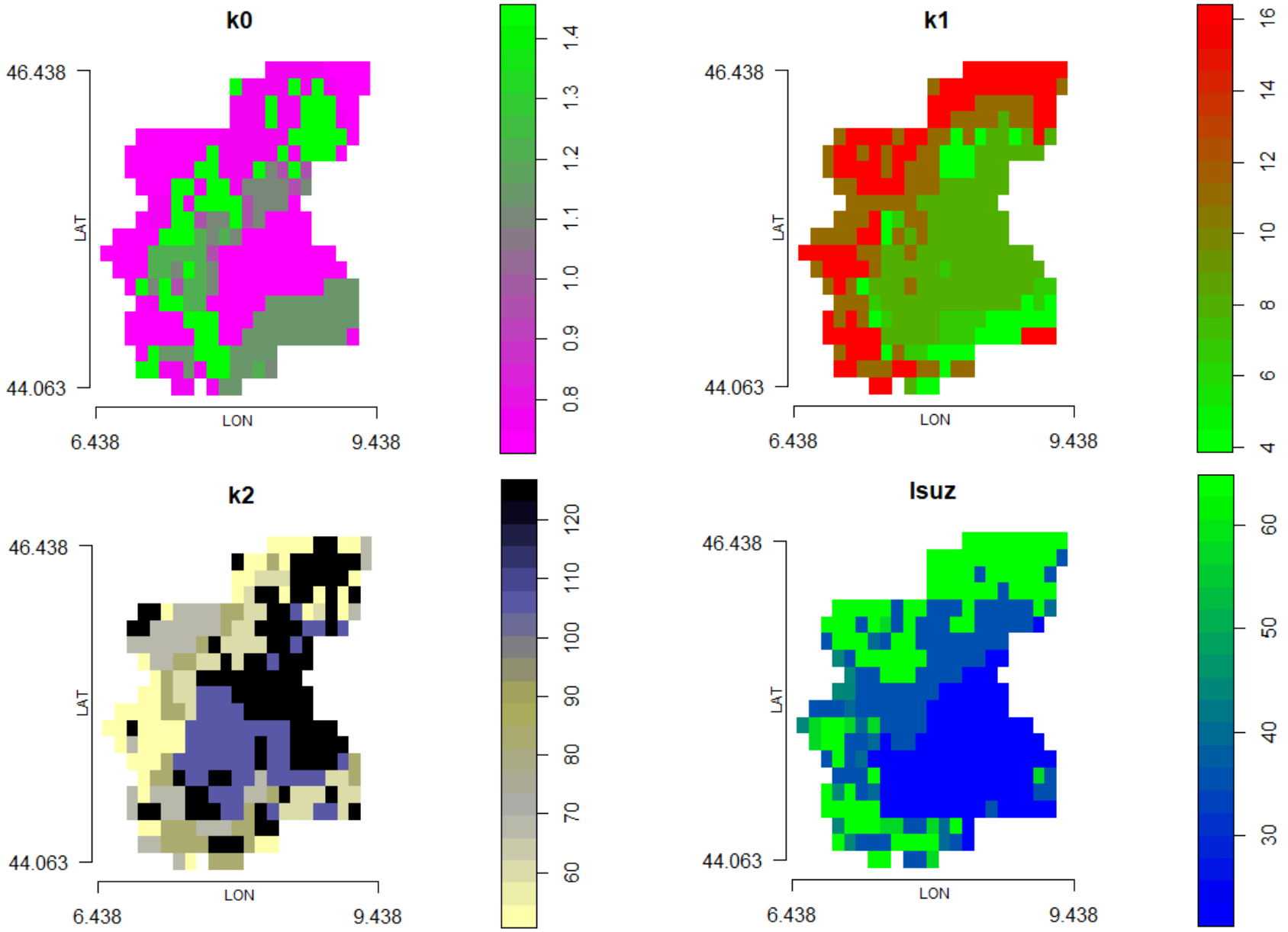


Figure 45 - Spatial map distribution (MODEL PARAMETERS) of k0, k1, k2, lsuz

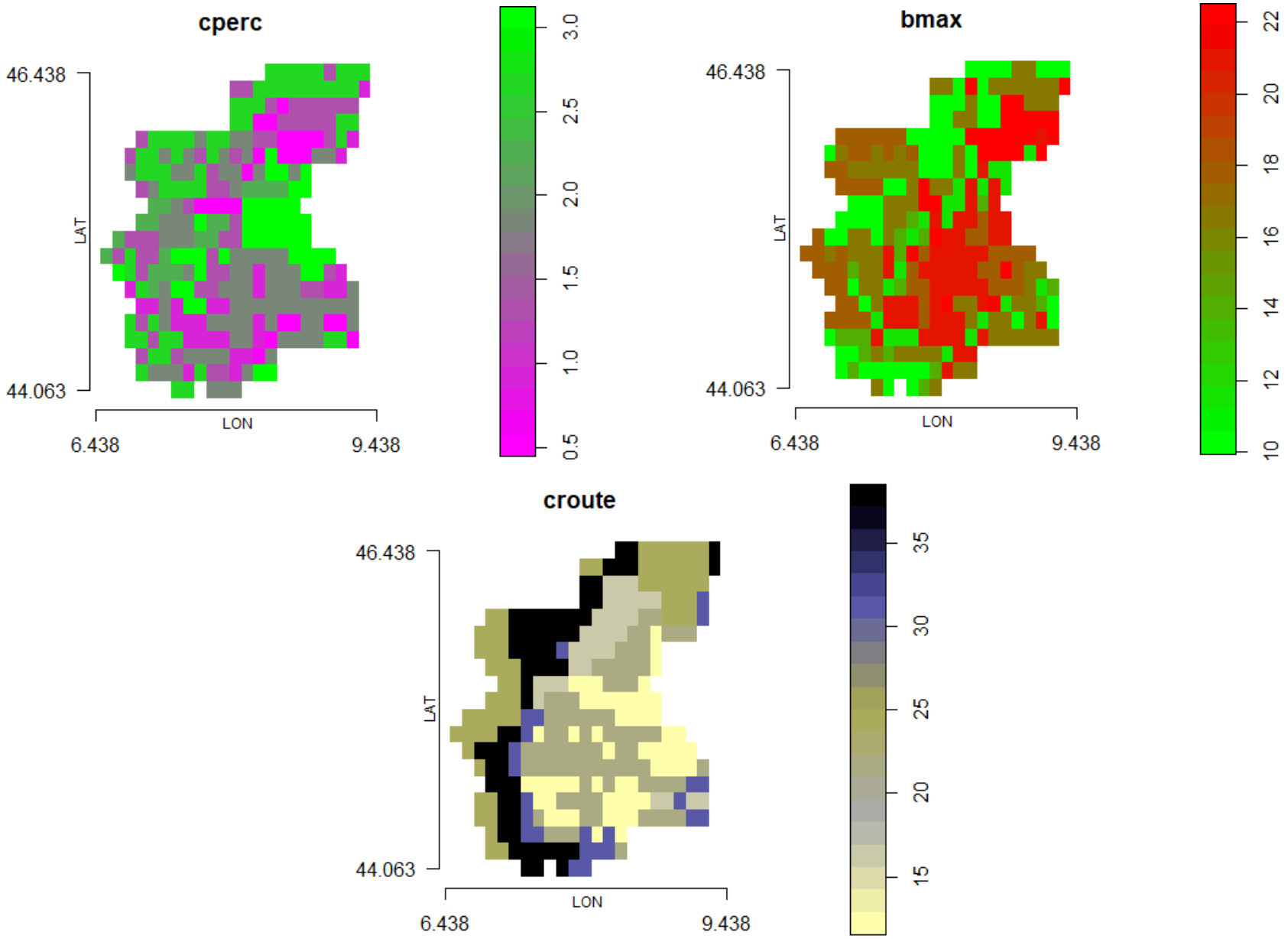


Figure 46 - Spatial map distribution (MODEL PARAMETERS) of cperc, bmax, croute

6.4 Validation

At the beginning of this thesis, in paragraph 1.2, the different steps of a regionalization procedure were listed, the last one being the validation process. A validation consists in verifying the temporal and spatial consistency of the obtained regionalization outside of the locations and years considered in the analysis to be sure of the robustness of the model obtained. It is a fundamental step for any application at ungauged sites: without it, it may be possible that the distributed model performs well only at gauged locations, but just outside of those areas, it completely misses out the predictions. In this work a specific validation procedure has not been conducted, but since an application is presented in the following chapter, some indications of the robustness of the *HydroPASS* procedure are here provided anyway, despite being obtained on partially different datasets and conditions. During the same period in which this thesis has been developed, a similar study has been conducted by Doctor Matteo Pesce at the *DIATI* department of Polito for his *PhD* thesis (still not published); the study area is the same with the same reference grid, but with some differences in the data. Together with the climatological data from the *OI*, additional information about snow cover were considered from *MODIS* product form terra and *Aqua* satellites. The same hydrological model (*TUWmodel*) was used, but with a different objective function accounting not only for the *KGE*, but also for a snow cover efficiency index. To test the spatial consistency of the regionalized model, the available gauged catchments were subdivided into two subsets; training catchments and verification catchments: the first were all the available catchment with an area lower than $1000\ km^2$ and with at least 5 local lumped parameters sets with $ME > 0.75$, used for the regionalization procedure with *PASS* (identical as the one described in this work). The remaining gauged catchments (validation catchments) have been used for the spatial validation. To test instead the temporal consistency, two different time frames have been considered: a calibration period from 2010 and 2020 (used for the regionalization), and a verification period from 2000 to 2010. The last difference regards the *CDs* used to determine the *RFR*: the 27 *ETCCDI* indices have been additionally introduced, but no *CN* have been used instead, for a total of 79 *CDs*. The results of the validation process are reported inside the article *"Regional multi-objective calibration for distributed hydrological modelling: a decision tree-based approach"* (Matteo Pesce et al., 2022) submitted to the *IAHS* (*International association of hydrological science*) but still under verification and not published yet:

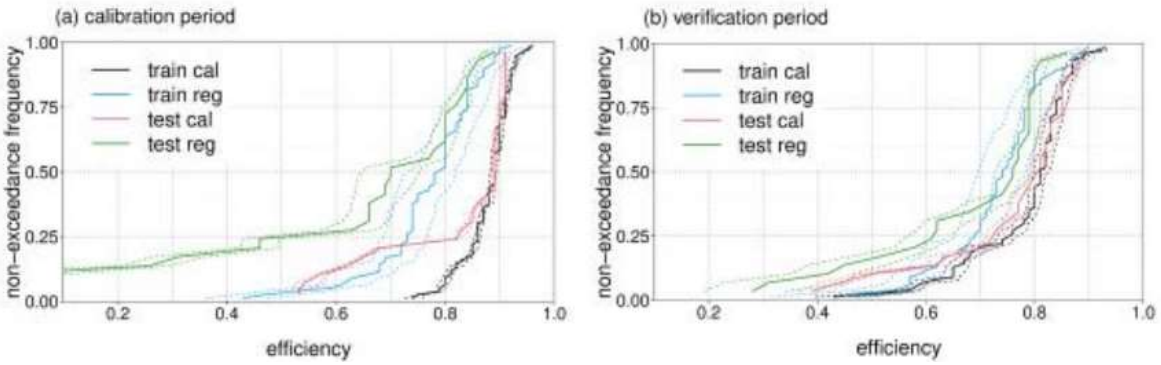


Figure 47 - Model efficiency eCDF for calibration (a) and verification (b) periods for train and test catchments (Pesce at al., 2022)

As shown in Figure 47, considering the calibration period, in local calibration the median of the training and test catchments *MEs* was nearly the same, but after regionalization it was around 0.785 for the training and decreases to 0.700 for the test catchments. Moving to the verification period instead, the median of the distributed efficiency lowered to 0.740 for the training basins and increased to 0.760 for the test ones. In general, they observed how for training and test catchments the median of the *MEs* were very close, with no significant degradation of the model performance between the two timeframes considered. "*These results confirm the robustness of the methodology in parameter estimation across spatial and temporal scales" (*Pesce at al. 2022).

The results of this study can be also used to justify, even if not prove, the following reported application for flood reconstruction. The use of different catchment descriptors, as shown in paragraph 6.3, probably implies slightly differences in the overall results. The basins used inside the validation procedure also come from the same database, with few differences. The main issues may be the absence of snow data (we could expect worst results for a validation process without them), and the timeframe considered, being our three times larger. Nevertheless, the *HydroPASS* regionalization procedure has been proven to be flexible and robust, giving good optimism about the significance of the simulations reported in the following chapter 7.

7. DISTRIBUTED MODEL APPLICATION AND FLOOD RECONSTRUCTION

One of the possible applications of the distributed model obtained after the regionalization is the reconstruction of past events by simulating discharges at ungauged locations. The idea is similar to the *PUB* (Prediction ungauged basins) hydrological studies with the main difference that with machine learning techniques multiple parameters sets for each catchment are used for the determination of the RFR instead of just the best one. The main idea is to start from a great number of basins without observation to reconstruct a complete map over the regional territory of flooding events. Other choices different from floods could have been made; the decision has been guided by the extremely relevance of such events and their impact on the environment surrounding the river network. Additionally, it has been considered of interest the possibility to reconstruct complete maps of such episodes, especially for historical floods, when the monitoring network was smaller and less widespread (moreover in Italy, after the Law Bessanini of 1998, the monitoring network passed from the national to the regional level, generating holes in the data especially in the 90s).

7.1 Data sources and catchment shape file manipulation

The data used for this application have been obtained from the *JRC* (European Joint Resource Center) and consisted of two distinct collections: the first one is a collection of Italian basins shape files, the second a collection of river channels shape files related to the catchments of the first one. Being the data about the whole Italy, a first subset has been created to reduce the analysis only on the Piedmont region. This operation has been performed using again the *R* package *"sf"* already utilized previously in chapter 3. The same function *"intersection"* has been applied. The Region boundary vector file here used to intersect the *JRC* database has been directly downloaded from *Geoportale Piemonte.* This first operation has been conducted only on the basins; the river network has been modified differently in a later step. The *JRC* data were in Lambert reference system, and so have been reprojected in *UTM32* to be compliant with the project reference system. The map of the resulting basins is reported in Figure 48:

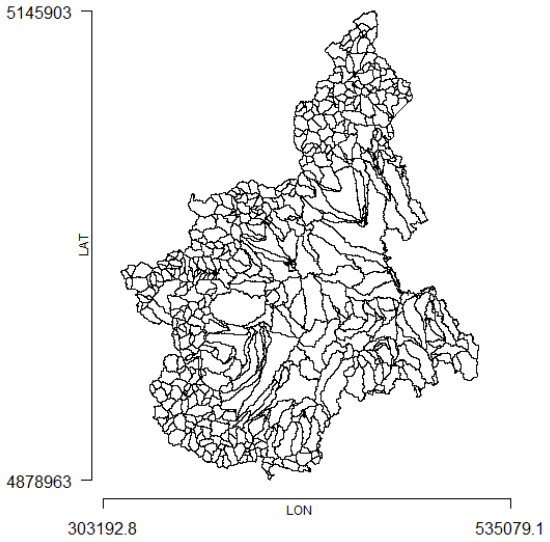


Figure 48 - JRC catchments Piemonte (UTM32)

The density of the basins is way higher than of the one of the original shapes used for the local calibration and regionalization (here there are 638 distinct catchments) to have a capillary, and so more representative, reconstruction of the historical events.

Before moving on, a second check must be done on the basins: for a correct application of the distributed model, we must be sure that all 638 catchments are contained inside the reference grid over which the distributed parameters have been calculated by the *HydroPASS* algorithm. If we add the grid to the plot, it can be immediately noticed how few catchments close to the boundaries of the grid are partially outside the external pixels:

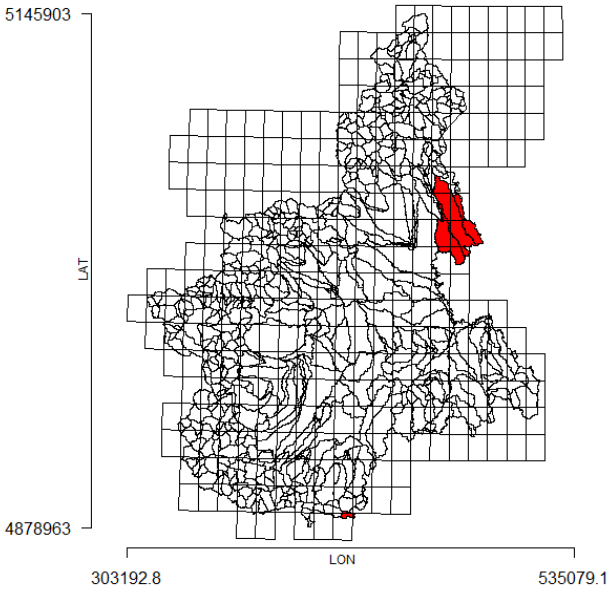
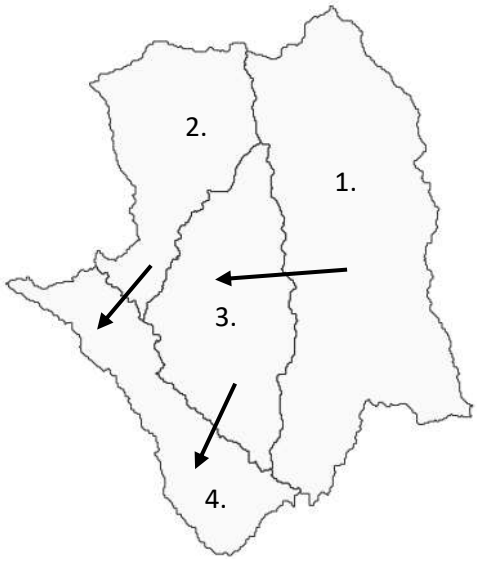


Figure 49 - JRC basins (in red partially external to the grid) (UTM32)

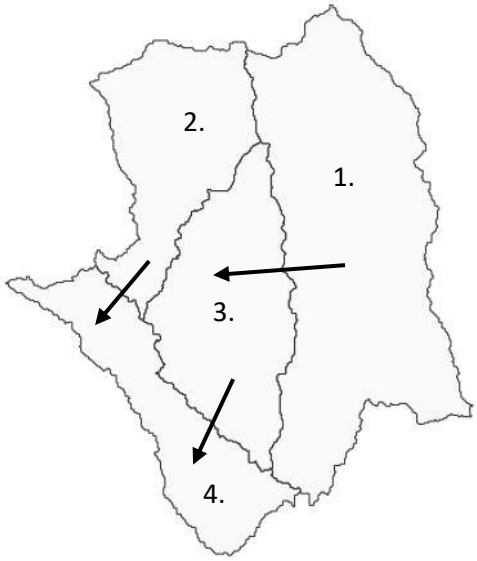
They have been removed for a new total of 633 different catchments. Remembering that our goal is to run *TUWmodel*, there is still the final and major problem to be addressed: as explained in chapter 3, *TUWmodel* does not account for inter-basin water routing, and so admissible catchments for the simulations are the ones that satisfy the conditions reported in $eq.3.5$. The *JRC* catchments structure on the contrary imply the transfer of water downstream. To solve this issue, a variable from the attribute table of the catchments provided together with the geometry data, where additional information about each basin of the dataset is stored, has been exploited. The *NextDown_ID* attribute specifies the basin immediately downstream of each outlet, defining the overall inter-basin flow path. A new geometries collection is determined using the following procedure:

1. A catchment is selected.
2. It is checked if the *ID* attribute of the selected catchment corresponds to the *"NextDown_ID"* attribute of any other catchment of the subset. One of two possibilities can verify:
   a) The *ID* of the selected catchment does not correspond to any of the *"NextDown_ID"*; it means that it is an upstream catchment that does not receive water by any other basin, but only transfers water downstream. In this case the basin geometry is added unmodified to the new collection.
   b) The *ID* of the selected catchment corresponds to at least one of the *"NextDown_ID"* attributes of the other basins; in this case a vector is created, and the *IDs* of all the immediately upstream catchments found this way are added to it. Now iteratively the search for upstream basins for the new *IDs* is performed until an iteration when all the *IDs* added to the vector from the previous one fall in case *A* is reached. The new geometry (with associated the *ID* of the original selected one at the first iteration) is created merging all the areas of all the upstream basins found during the process.
3. The procedure is repeated for all the basins.

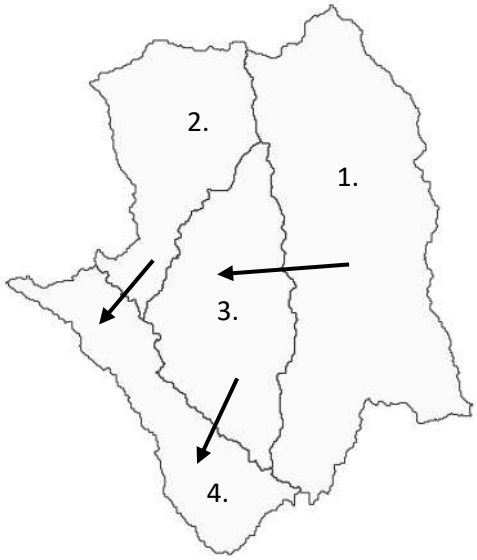
It follows a schematic example to better visualize the strategy:



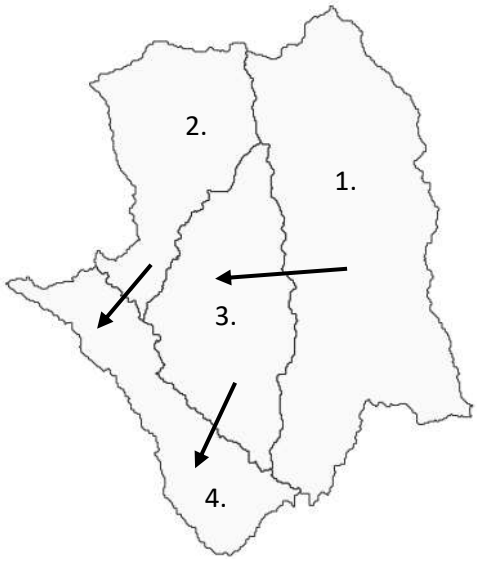
Basin 1 does not receive water from upstream, thus the new geometry is:



Basin 2 does not receive water from upstream, thus the new geometry is:



Basin 3 receives water from basin 1, thus the new geometry is:



Basin 4 receives water from basin 2 and 3. Basin 3 receives water from basin 1. Thus, the new geometry is:

For the application of *TUWmodel*, as usual, together with the climatological data, we also need a new *"train topology"* for the new catchments just defined. It has been obtained analogously to what have been already done in chapter 3.

The second subset (the one for the river network) has been created exploiting again another variable in the table of attributes of each basin and each river channel. The *"ZHYD"* attribute (indicating the identification number of the *FEC – functional elementary catchment*) has been used to extract from the original river databases only the sections contained inside at least one of the basins belonging to the newly obtained subset. The result is shown in Figure 50:

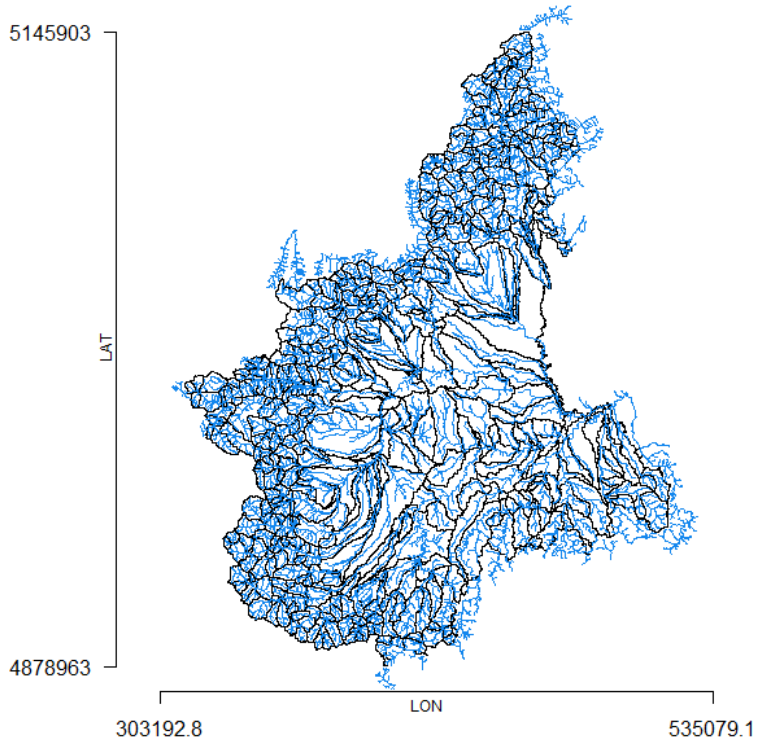


Figure 50 - JRC basins and river network over Piemonte (UTM32)

Of course, a denser river network is present on the mountainous areas where we are in a torrential environment while bigger rivers form downstream in the plains. It can be noticed how some rivers exceed the regional boundaries: this happens since in the database a river channel with a *FEC ID* may run also over different catchments, in these cases external ones. Simply cropping the network with the same region boundary shape file of *"Geoportale Piemonte"* already used above solves this graphical problem.

7.2 Historical flood reconstruction

The simulations have been realized selecting specific 6-days periods where historically a flooding event struck the Piedmont region using the distributed parameter from Group 5, the one with best average regional *ME*, as presented in chapter 6. Five events have been reconstructed: the flood of 1994 (3rd of November – 8th of November), of 2000 (13th of October – 18th of October), of 2008 (27th of May – 1st of June), of 2016 (22nd of November – 27th of November), and of 2020 (30th of September – 5th of October). The colors in the maps, like in Figure 52, have been assigned to the river channels depending on the simulated generated runoff values (mm/d) with a color scale form light blue (low discharge) to red (high discharge). The descriptions of the events have been taken from the official reports downloadable from the *ARPA* Piemonte website (see bibliography).

Flood 1994

The flood in 1994 is today still considered as one of the most severe events in the history of Piedmont. The precipitations, initially continuous and of low intensity, started on the 2nd of November and continued in the following two days; on the 4th of November a great intensification started, interesting majorly the southern part of the region (close to the Ligurian Apennine), the municipality of Biella and the Langhe area. From a posterior analysis of the fallen precipitation three main phases of the event have been identified:

- The first phase has interested mainly the south, close to the border with the Liguria region, on the 4th of November; peak precipitation over $150\ mm$ with intensities over $35\ mm/h$ (Ponzone, Alessandria) had been recorded by the fixed monitoring stations. Simultaneously intense precipitation had also been registered in the north, but with lower intensities and cumulative values around $100\ mm$. In the morning of the 5th of November, the precipitations reduced of intensity in the south.
- During the second phase, on the 5th of November, the precipitations extended over the districts of Cuneo and Asti where the most struck basins have been the Tanaro, Bormida and Belbo. In the first hours of the morning (from 2.30 A.M. to 7.30 A.M.) the precipitation intensity had never gone below $20\ mm/h$, while in the remaining of the morning they remained around 10-$15\ mm/h$ to then again increase in the afternoon with peaks of over 35 mm/h. After 9 P.M. the precipitations nearly stopped over all this area. Overall, more than $200\ mm$ (in some case closer to 250) have fallen in less than 24 hours.

- The third and last phase involved mainly the Sesia and Pellice Valleys during the afternoon and the evening of the 5th of November, to reduce of intensity during the morning of the following day. On the 6th of November the precipitation completely stopped in the south, continuing with lower intensities, but relevant cumulative values in the northern sector of the region.

Overall, such precipitation events had been considered exceptional, greatly overcoming the reference values all over the involved areas. On over around 40% of the regional territory, in the timeframe from the 5th to the 7th of November, has fallen more than $200\ mm$ of rainfall.

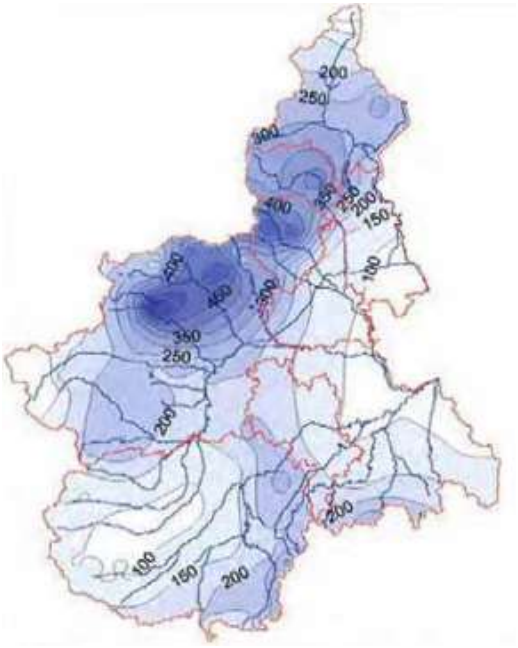


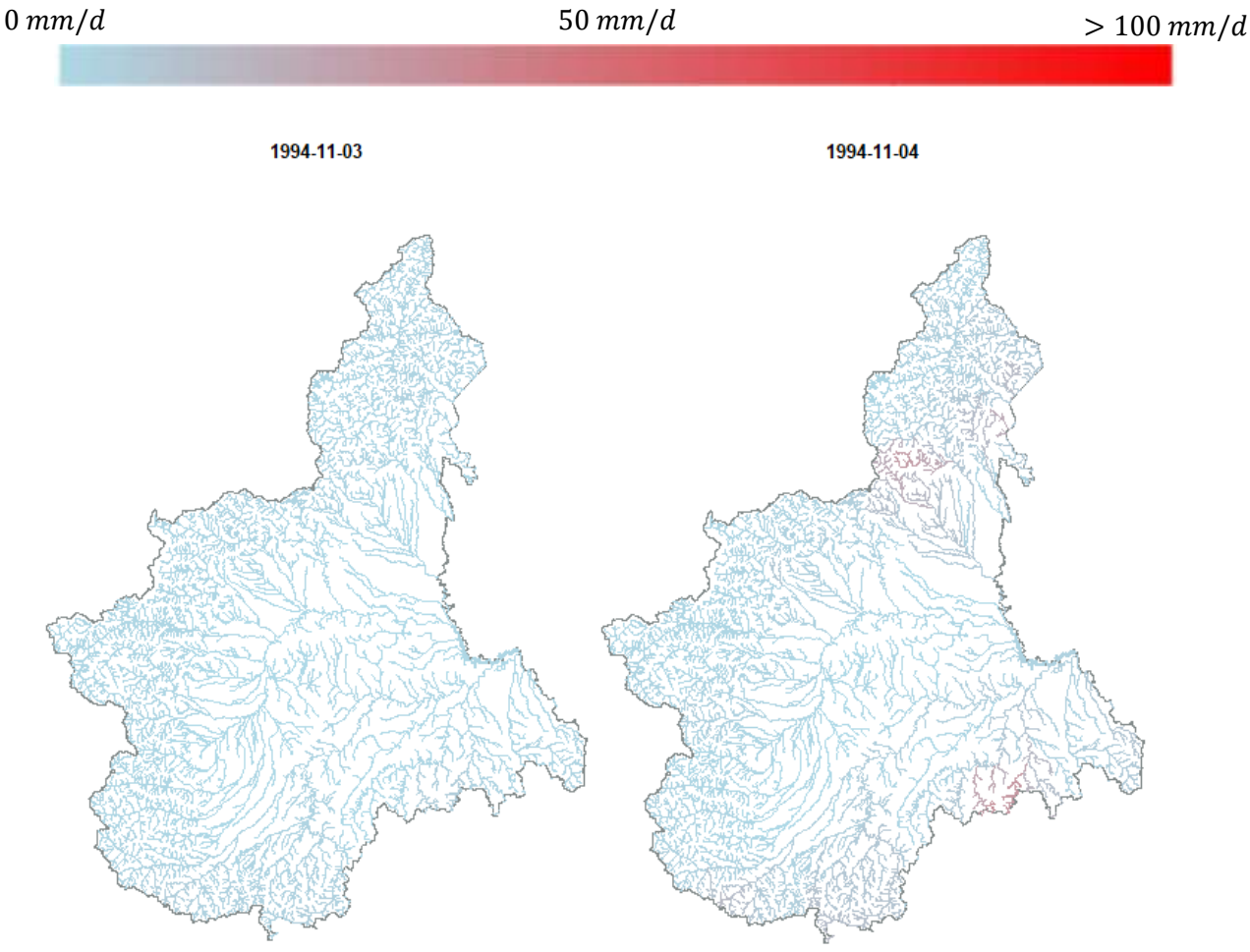
Figure 51 – Isolines analysis of the observed cumulative rainfall from 4th of November 00 UTC to 7th of November 00 UTC ("Eventi alluvionali in Piemonte 2-6 Novembre 1994, 8 Luglio 1996, 7-10 Ottobre 1996", Vincenzo Coccolo et al., 1998)

The intense precipitations directly transformed in large discharge values in the river network, with subsequent flooding events in the surrounding areas: the basins in the south registered discharges (in some cases they have been only estimated) coherent with events with a return period of over 200 years, for the western sector of the region with events with a return period of around 20 years, and for the north sector with event having a return period of 20 years (for the Sesia Basin) and of 200 years (for the Dora Baltea Basin, after the confluence of Chiusella). A predisposing cause partially explaining such values were the conditions of the soil before the beginning of the intense rainfalls: in fact, in the second part of the month of October 1994, frequent precipitations occurred, creating high saturation conditions that helped the new precipitations to transform immediately into effective

runoff and to increase the water level. This fact can also be seen by the average water level at some stations on the 3rd of November before the event if compared with the average reference values as reported in Table 16:

Stazione	*Sup. bacino (km²)*	*Portata 3-11-94 (m³/s)*	*Portata media annua (m³/s)*
Po a Cardè	510	10	11
Po a Carignano	3804	85	71
Po a Torino	5210	90	95
Chisone a S. Martino	581	9	15
Dora Riparia a Susa	628	19	11
Stura di L. a Lanzo	582	30	20
Dora Baltea a Mazzè	3837	58	96
Sesia a Borgosesia	695	80	33
Stura di Demonte a Gaiola	562	20	18
Tanaro ad Alba	3415	80	71

Table 16 - Hydrological characteristics of some significant stations ("Eventi alluvionlai in Piemonte 2-6 Novembre 1994, 8 Luglio 1996, 7-10 Ottobre 1996", Vincenzo Coccolo et al., 1998)



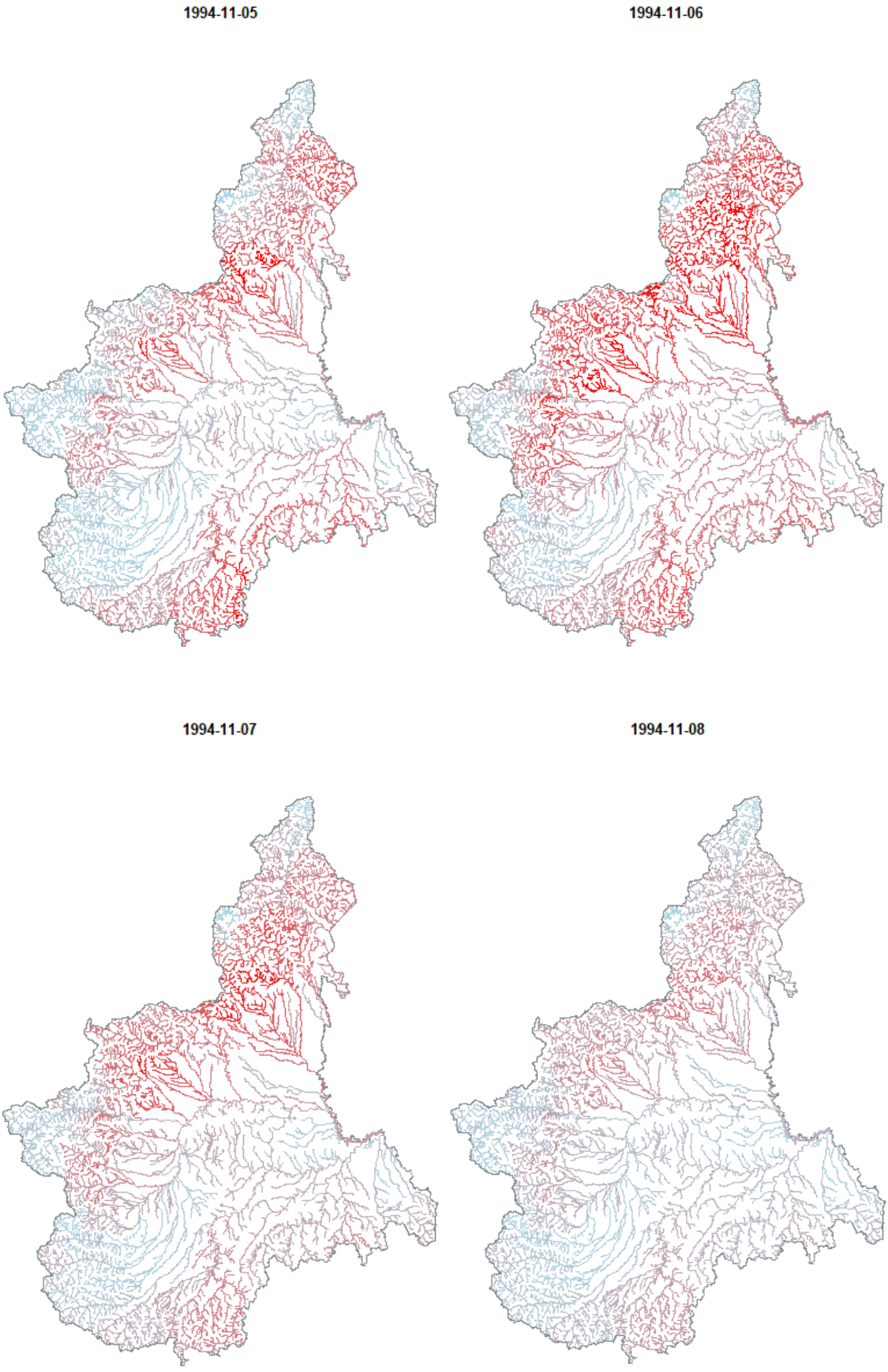


Figure 52 – Regional simulation (TUWmodel) using the HydroPASS regionalized parameters of the flood event in 1994

Despite it is impossible to directly compare the simulated values with observation at the ungauged locations (no monitoring system installed), it can be observed how the reconstructed discharges are coherent with the historical description of the event and of the most damaged areas. Also, the temporal consistency seems to be respected: initially the highest discharges have been simulated in the south, then increasingly on the north, and reduces first on the south on the 7th and then on then north on the 8th.

Flood 2000

The flood in the year 2000 (from the 13th to the 17th of October), differently from the one in 1994, interested mainly the north and west sectors of the region, affecting the southern part with a reduced intensity. At the beginning of the event, on the 13th of October, the first intense rainfalls involved the Verbano Occidentale and Sesia Valleys and Canavese area; nevertheless, most of the intense precipitations concentrated during the 14th and 15th wide spreading all over the Pellice-Po Valley, high Tanaro Valley, and Susa-Sangone Valley. The second part of the event, differently from the first, was partially influenced by a fraction of solid precipitation on the Alpine sectors, that contributed to slow down the generation of effective runoff and the hydrological response of the river network. Looking instead at the intensity of the precipitations, it resulted considerably high both for short (1,3, and 6 hours) and long durations (12, and 24 hours). Generally, in the struck areas the intensity never went below $10\ mm/h$, with frequent peaks over $20\ mm/h$ and in some cases above 40-$45\ mm/h$. Moreover, only in few locations the precipitation was discontinuous, being instead uninterrupted everywhere else.

Considering the entire duration of the event, the highest cumulated rainfalls were recorded in Verbano Occidentale ($747\ mm$ at the Bognanco Pizzanco station, $732\ mm$ at the Bognanco Lago Paione station), Sesia Valley ($665\ mm$ at Boccioleto Ronchi station), Canavese and Lanzo Valleys ($716\ mm$ at Ala di Stura station, $698\ mm$ at Piamprato station), and Sangone and Po Valleys ($598\ mm$ at Coazze station). In the Tanaro Valley, in the south, the highest peak was considerably lower ($284\ mm$ at Briga Alta-Piaggia station).

Moving instead to the hydrological response of the river network, considering the large areal extension of the intense precipitations, all the left tributaries of the Po River, till the Ticino River, experienced exceptional discharges and floods. The peak discharge of the Chisone River overcame the estimated values for a return period of 100 years and exceptional resulted also the value for the Sangone River, but no data were there available due to the failure of the monitoring station. In Turin the peak discharge was recorded in the morning of the 16th of October at a value of 2350 m^3/s, higher

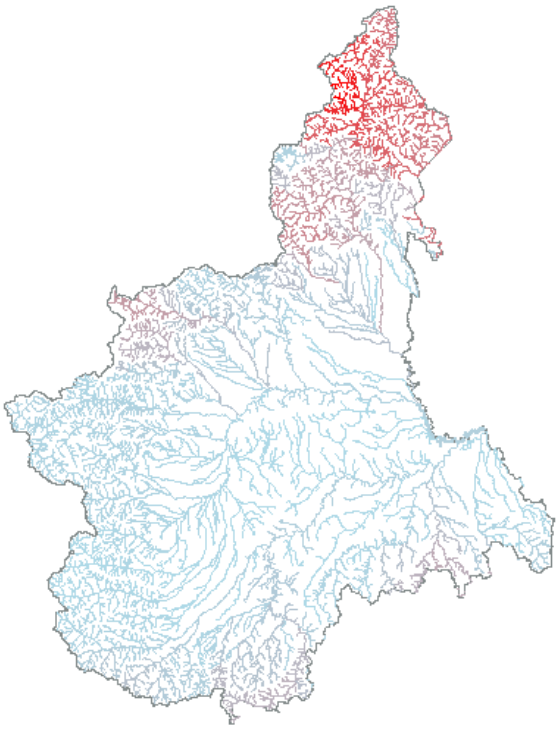
than the previous historical maximum of $2230\ m^3/s$ recorded during the flood in 1949. Downstream of the city the discharge values increased due to the contribution of the Dora riparia, Stura di Lanzo, Malone and Orco Rivers. Also, for the Dora Riparia the recorded values represented one of the strongest ever recorded events with a peak of $700\ m^3/s$ before the confluence with the Po River. The Tanaro Basin presented instead lower values than the one recorded during the flood in 1994, with peak values in Montecastello of $3000\ m^3/s$ (in 1994 it was of $4800\ m^3/s$). Some summarized hydrological information is reported in Table 17:

Stazione	Colmo [m]	Data [UTC]	Incrementi massimi registrati [m]					
			1 ora	3 ore	6 ore	12 ore	24 ore	Totale
Po a Carignano	6.29	04:00 16-10-00	0.36	0.75	1.33	2.55	3.94	5.26
Po ai Murazzi	5.79	11:30 16-10-00	0.36	0.96	1.81	2.74	4.29	5.49
Po a Crescentino	6.45	04:00 16-10-00	0.57	0.73	1.28	1.86	3.03	4.74
Po a Isola S_Antonio	9.31	15:00 16-10-00	0.23	0.64	1.25	2.07	3.29	7.72
Po a Casei Gerola	5.17	16:00 16-10-00	0.22	0.63	1.16	1.97	3.7	7.05
Po a Ponte Becca	7.81	04:30 17-10-00	0.17	0.51	0.94	1.74	3.03	8.12
Chisone a S.Martino	4.05	14:30 15-10-00	0.80	1.32	2.12	3.32	4.32	4.72
Dora Riparia a Oulx	1.77	03:00 15-10-00	0.29	0.72	0.98	1.22	1.33	1.42
Cenischia a Susa	2.97	19:30 15-10-00	1.84	2.41	2.48	2.54	2.55	2.82
Ceronda a Venaria	3.07	21:30 15-10-00	0.96	1.56	1.71	2.24	2.3	2.67
Malone a Front	2.69	07:30 15-10-00	0.69	0.92	1.25	1.35	1.86	2.19
Orco a Cuorgnè	4.29	12:30 14-10-00	1.07	1.13	2.01	2.10	3.55	3.78
Soana a Pont Canavese	4.28	03:30 15-10-00	1.13	1.13	1.28	1.78	2.45	3.23
Sesia a Borgosesia	4.87	06:00 15-10-00	1.14	1.77	2.5	2.71	3.17	5.71
Mastallone a Varallo	4.6	16:30 13-10-00	1.06	1.58	2.28	2.78	2.78	3.24
Diveria a Crevola D'ossola	4.02	08:00 14-10-00	0.53	0.93	1.55	1.9	2.28	2.7
Bogna a Pontecaddo	4.57	16:30 13-10-00	1.46	1.65	2.4	2.75	2.75	3.19
Toce a Candoglia	9.16	16:30 15-10-00	0.62	1.79	3.24	4.83	6.61	7.57
Lago Maggiore a Verbania	7.94	21:30 16-10-00	0.08	0.17	0.33	0.6	1.15	3.56
Tanaro a Farigliano	4.7	08:00 15-10-00	0.5	1.2	1.98	2.99	3.25	3.50
Tanaro Ad Alba	3.47	00:30 16-10-00	0.47	1.03	2.03	3.02	3.33	3.57
Tanaro a Montecastello	6.92	17:00 16-10-00	0.88	2.39	3.81	4.79	5.21	6.28

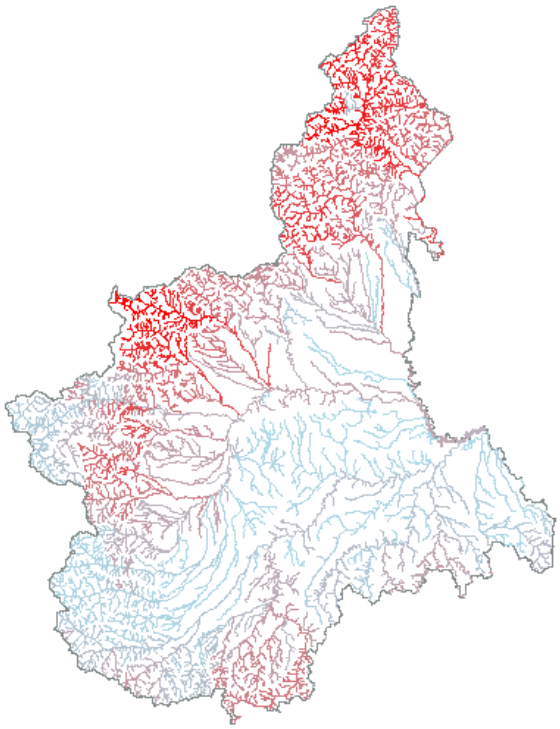
Table 17 – Synthesis data relative to the more relevant hydrographs recorded between the 13th -16th of October 2000 ("Evento alluvionale del 13-16 ottobre 2000 in Piemonte Analisi meteorologica e idrologica", Direzione Regionale Servizi Tecnici di Prevenzione, 2001)

$0\ mm/d$ $50\ mm/d$ $> 100\ mm/d$

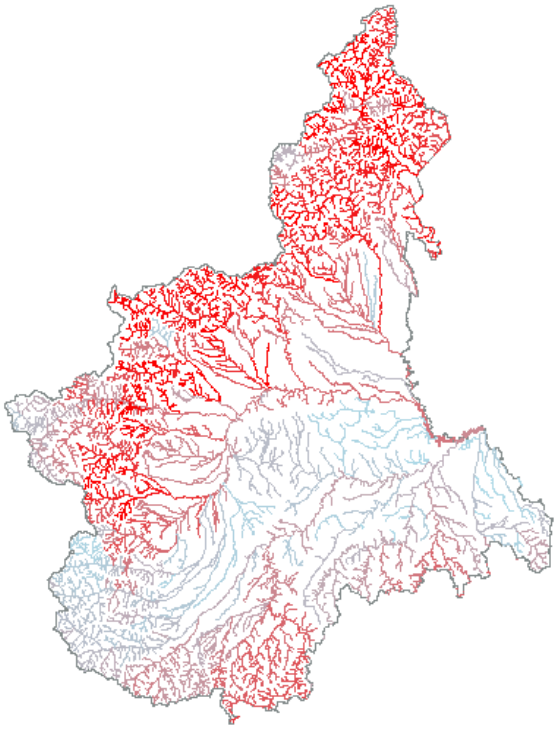
2000-10-13



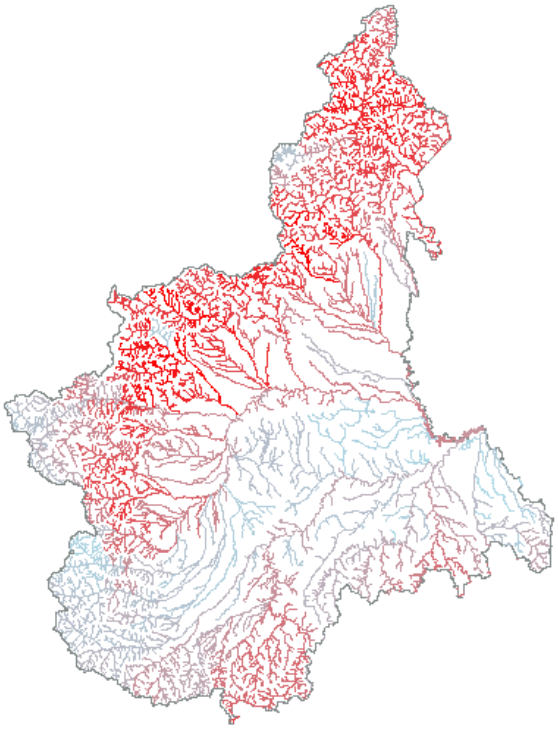
2000-10-14



2000-10-15



2000-10-16



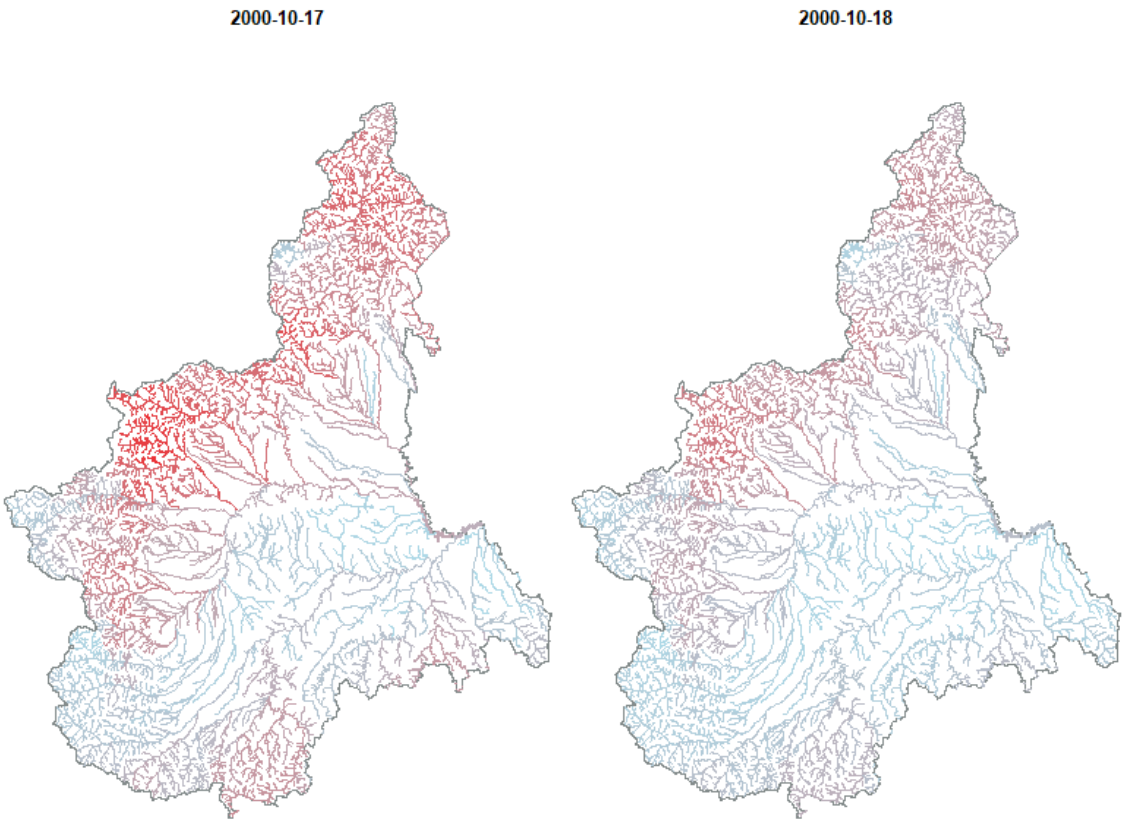


Figure 53 - Regional simulation (TUWmodel) using the HydroPASS regionalized parameters of the flood event in 2000

Again, also for this second case the regionally simulated discharges seem to be coherent with the description of the historical event. The different magnitude between the west-north and south sectors is well reconstructed as well as the timing, with the most intense response concentrating on the 15th and 16th of October.

Flood 2008

The third event here described developed back in end of May of 2008, starting from the 27th till the 30th of the month. Similarly to what happened during the flood of 1994, the intense precipitations during the event can be subdivided in three main phases:

- During the first phase (relative to the 27th of May), the rainfall was generally moderate but locally intense in the Orco, Stura di Lanzo, Dora Riparia, Stura, and Pellice Basins (with peaks of $112.2\ mm$ fallen at Lago Agnel, Ceresole Reale, and $109.2\ mm$ at Rifugio Gastaldi, Balme).

- After an interruption during the morning and afternoon of the 28th of May, during the evening of the same day the second phase began lasting for all the 29th: the areas interested by the event extended to the municipality of Cuneo, while in the same basins of the first phase, it increased in intensity: at Lago Agnel were recorded $155\ mm$ while at Rifugio Gastaldi $172.4\ mm$. In Grana and Stura di Demmonte Valleys (*CU*) even higher peaks were recorded in the localities of Castelmagno ($191.8\ mm$) and San Giacomo di Demonte ($177.8\ mm$).
- The third and last phase developed during the 30th of May. Here the intensity of the precipitation reduced, and the territory was interested only by residual rainfall.

An additional factor contributed to the high impact of the precipitation during the event: being in the summer season, the snow line was close to $3000\ m$; in other words, most of the precipitation fallen in liquid form, and only a reduced fraction was stored in solid form as snow. Overall, considering the entire duration of the event, many stations along the Alpine and pre-Alpine areas of Piedmont (the one more severely struck) recorded cumulated values higher than $200\ mm$, with highest peaks in the Pellice Valley ($425.8\ mm$ at Colle Barant) and Germanasca Valley ($336.8\ mm$ in Massello).

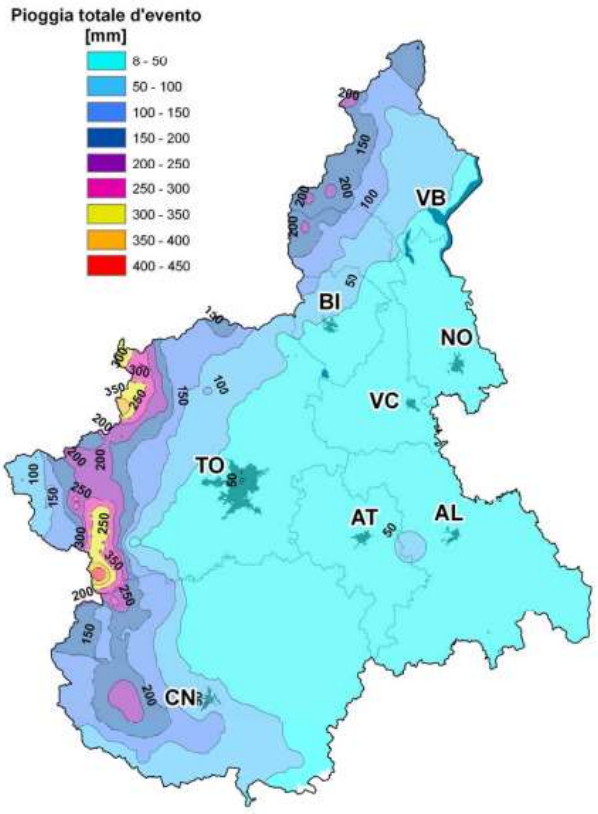


Figure 54 – Isohyets of cumulative precipitation for the entire duration of the flood of 2008 over the Piedmont region ("RAPPORTO PRELIMINARE SULL'EVENTO ALLUVIONALE DEL 28-30 MAGGIO 2008", ARPA Piemonte, 2008)

From a statistical point of view the experienced rainfall resulted particularly critical looking at the duration of 12 and 24 hours, where they often overcame the reference values relative to a return period of 50 years, sometimes approaching the value for 100 years.

Moving instead to the hydrological response, the rivers of the network more affected have been the one contained in the basins of the west-Alps, from the Susa to the Stura di Demonte Valley; here the highest criticalities took place. It is important to underline how in the period before the event (since April), continuous rainfalls interested the entire region, creating a partially saturate soil condition that favored the quick generation of effective runoff.

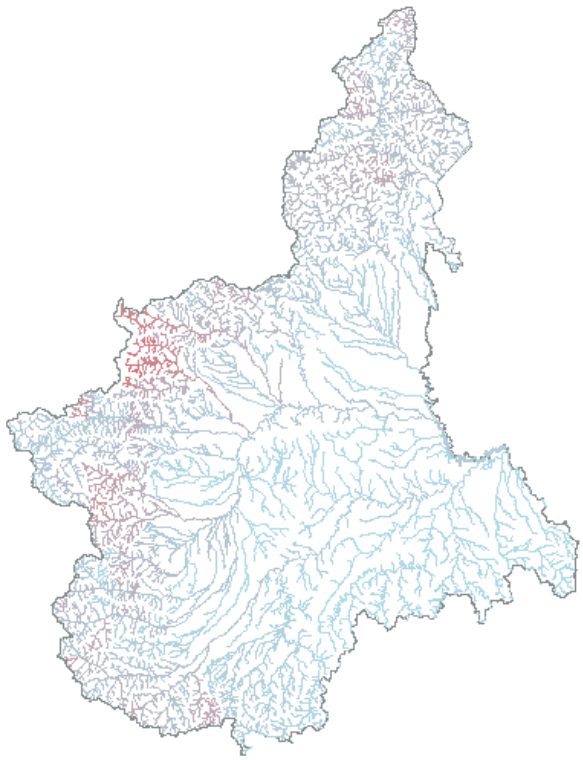
Moderate (but indeed more limited) problems also developed in the Lanzo, Canavese and Dora Baltea Valleys. The first severe overflows were recorded in the morning of the 29th of May along the Dora riparia, Germanasca, Chisone and Pellice Rivers. The intensification of the precipitation during the second phase caused the additional floodings along the Toce, Sesia, Dora Baltea, and Orco Rivers. Exceptional events interested the valley in the municipality of Cuneo, where the rivers' levels remained for an extended period above the critical values for the Varaita, Grana, and Stura di Demonte. Also in this case, the statistical analysis relative to the return period for the main affected location by the event are summarized in Table 18:

Bacino	ZONA	Stazione	Portata [m^3/s]	TR [anni]
Lago maggiore	Piem-A	PALLANZA VERBANO	-	-
	Piem-A	CANDOGLIA TOCE	900	<5
Sesia	Piem-B	CAMPERTOGNO SESIA	250	<10
Dora Baltea	Piem-B	TAVAGNASCO DORA BALTEA	1100	<10
	Piem-I	PARELLA CHIUSELLA	250	-
	Piem-I	VEROLENGO DORA BALTEA	750	<5
Orco	Piem-C	PONT SOANA	-	-
	Piem-C	CUORGNÈ ORCO	300	<5
	Piem-L	SAN BENIGNO ORCO	400	<5
Malone	Piem-L	BRANDIZZO MALONE	140	<5
Stura di Lanzo	Piem-C	LANZO STURA DI LANZO	600	5
	Piem-L	TORINO STURA DI LANZO	550	<5
Dora Riparia	Piem-D	OULX DORA RIPARIA	100	<5
	Piem-C	SUSA DORA RIPARIA	250	40
	Piem-L	TORINO DORA RIPARIA	400	40
Pellice	Piem-D	PERRERO GERMANASCA	-	-
	Piem-L	SAN MARTINO CHISONE	650	30
	Piem-M	VILLAFRANCA PELLICE	1100	20
Varaita	Piem-E	ROSSANA VARAITA	250	20
	Piem-M	POLONGHERA VARAITA	-	-
Maira	Piem-E	MONTEROSSO GRANA	-	-
	Piem-M	BUSCA MAIRA	193	20
Stura di Demonte	Piem-E	GAIOLA STURA DI DEMONTE	400	50-100
	Piem-M	FOSSANO STURA DI DEMONTE	500	50
Ellero	Piem-F	MONDOVÌ ELLERO	100	<5

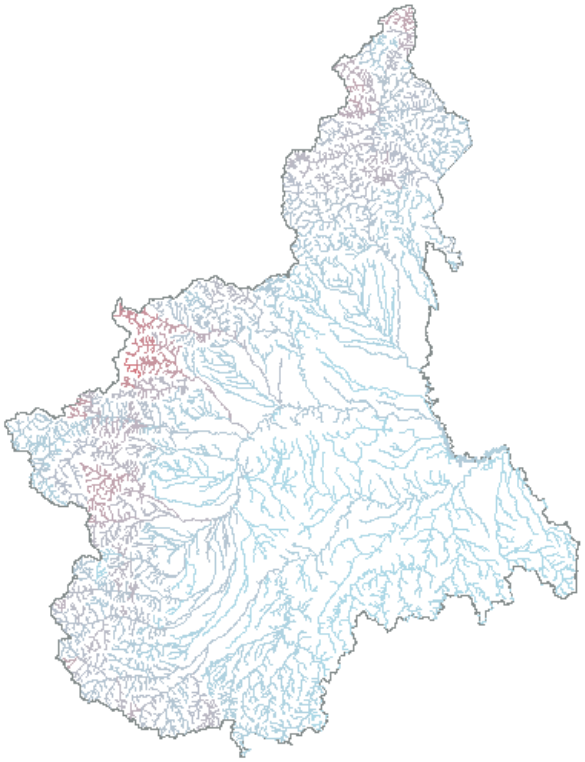
Table 18 -Synthesis data about the return period of the observed discharges in the most affected sections during the flood event in the November 2008 in the Piedmont region ("RAPPORTO PRELIMINARE SULL'EVENTO ALLUVIONALE DEL 28-30 MAGGIO 2008", ARPA Piemonte, 2008)

$0\ mm/d$ $50\ mm/d$ $> 100\ mm/d$

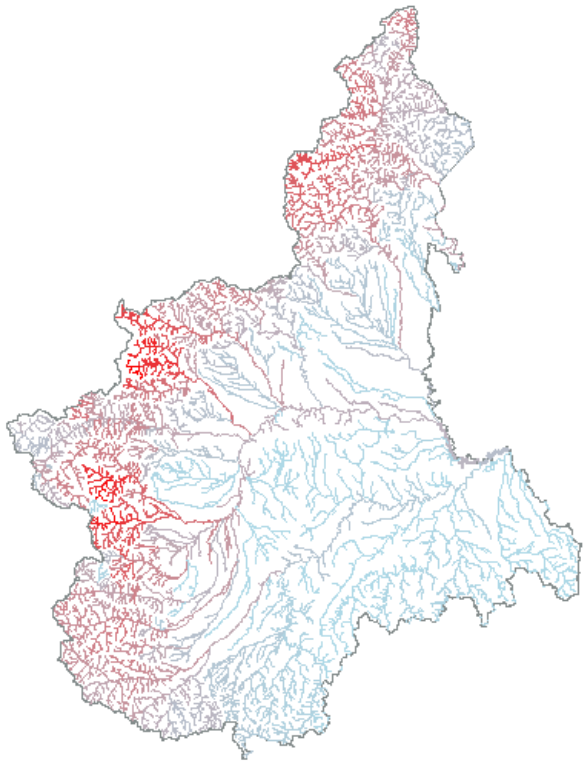
2008-05-27



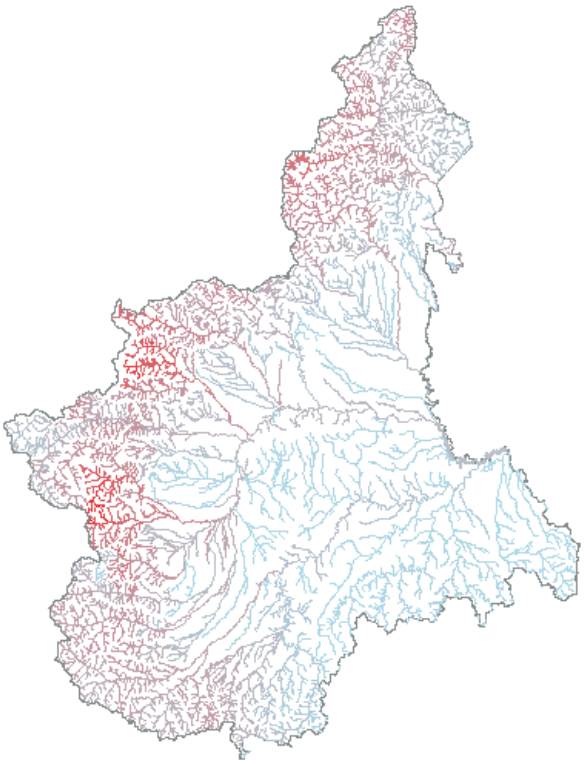
2008-05-28



2008-05-29



2008-05-30



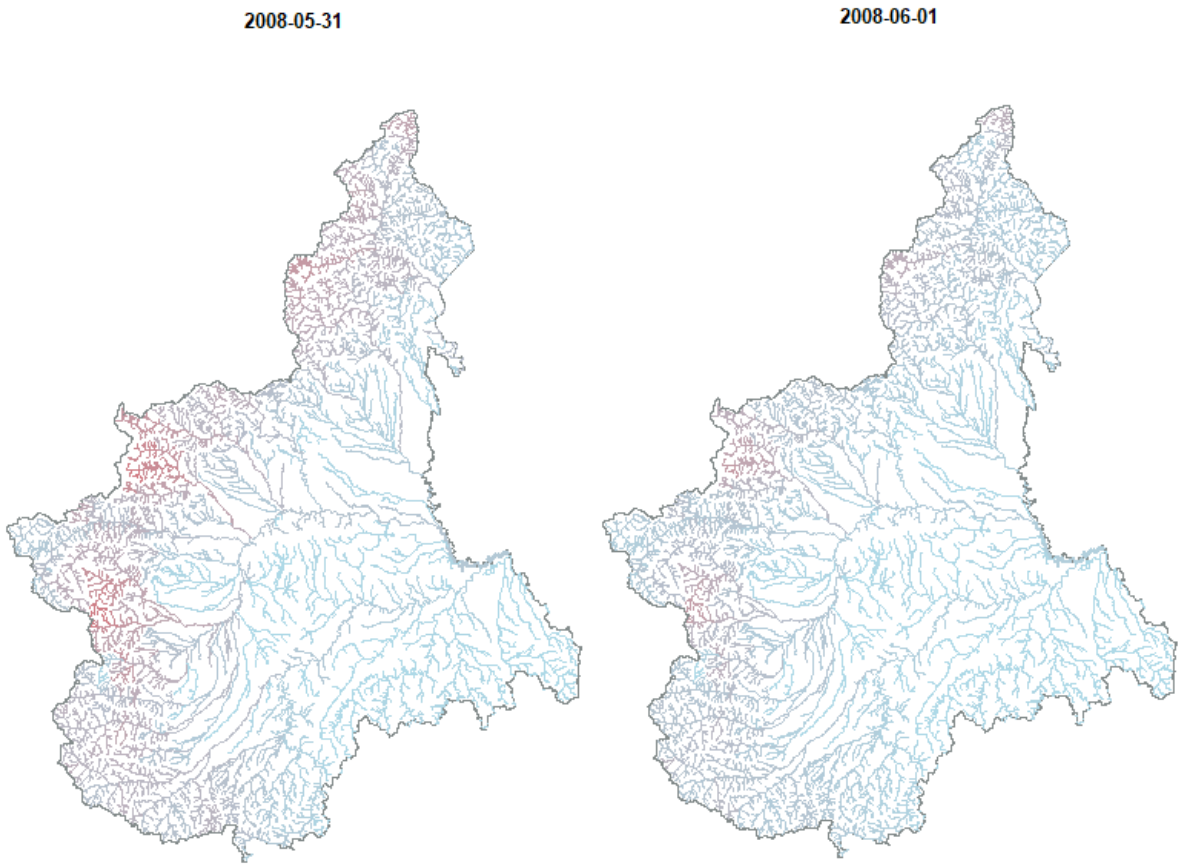


Figure 55 - Regional simulation (TUWmodel) using the HydroPASS regionalized parameters of the flood event in 2008

Good results have been obtained also in this third case: comparing the maps in Figure 55 with the map of cumulative rainfall during the event in Figure 54 it can be easily observed the good match in timing and position of the generated runoff.

Flood 2016

Between the 21st and 25th of November 2016 an intense precipitation event effected the Piedmont region interesting initially the municipalities of Cuneo and Alessandria close to the border with the Liguria region and afterward extending also to the municipalities of Biella, Vercelli, and Turin. The most intense rainfalls have been recorded on the 24th in the western sector of the region and in the south in the Tanaro Valley. The stations characterized by the highest precipitation recording have been Piaggia (*CU*, $633\ mm$), Ponte di Nava Tanaro (*CU*, $621\ mm$), and Calizzano (*SV*, $612\ mm$) in the high Tanaro Valley. In the high Po Valley, the rain gauge of the Barge (*CU*) station recorded $593\ mm$ while in the Stura di Lanzo basin $610\ mm$ were recorded by the monitoring station in Niquidetto (*TO*). These values represented for the single stations more than 50% of the total mean annual expected values.

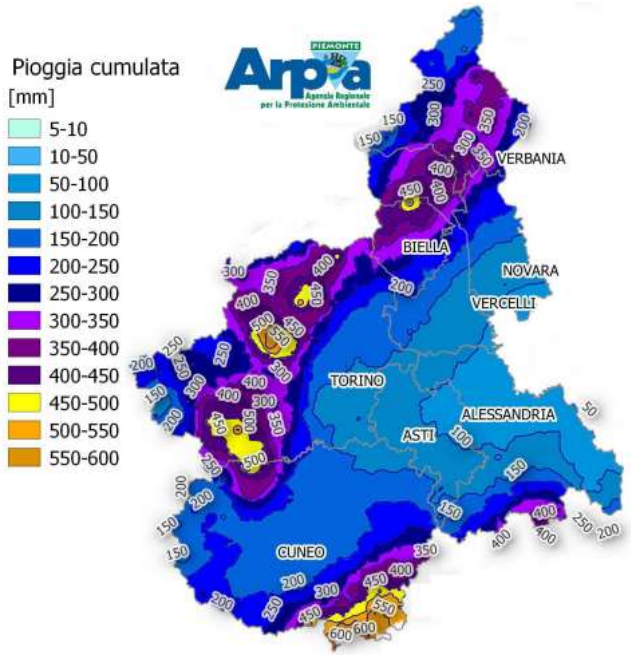


Figure 56 - Isohyets of cumulative precipitation for the entire duration of the flood of November 2016 over the Piedmont region ("Gli eventi alluvionali in Piemonte – Evento del 21-25 novembre 2016", ARPA Piemonte, 2018)

At the stations majorly involved by the event the estimated return periods of the observed precipitations overcame the reference values for 50 years for a duration of 24 hours. In general, the statistical analysis reflected the characteristics of an event which return period increased with the duration as shown by the graphs relative to a couple of the most affected station (Niquidetto and Piaggia) in Figure 57:

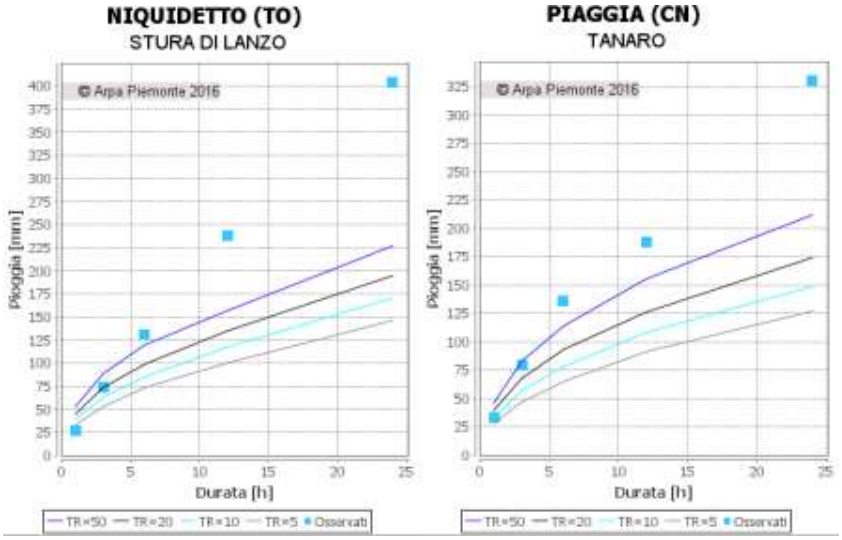


Figure 57 – Reference lines for return period of 5, 10, 20, 50 years compared to the observed data during the flood event of November 2016 for the stations of Niquidetto (TO) and Piaggia (CU) region ("Gli eventi alluvionali in Piemonte – Evento del 21-25 novembre 2016", ARPA Piemonte, 2018)

Differently for the other flood events here analyzed, the one of 2016 developed during a period when the snow line was never particularly high, enabling the formation of solid precipitation and deposition of snow over the montaneous areas. Initally the snow line was around $1600\,m$ on the 21st of November to rise till $2300\,m$ the next day till the 24th to then return around $1500\,m$. Snow precipitation initially concentrated on the north and south sectors ad only in a second moment extended to the all region but with a lower intensity to then increased again on the wester sectors:

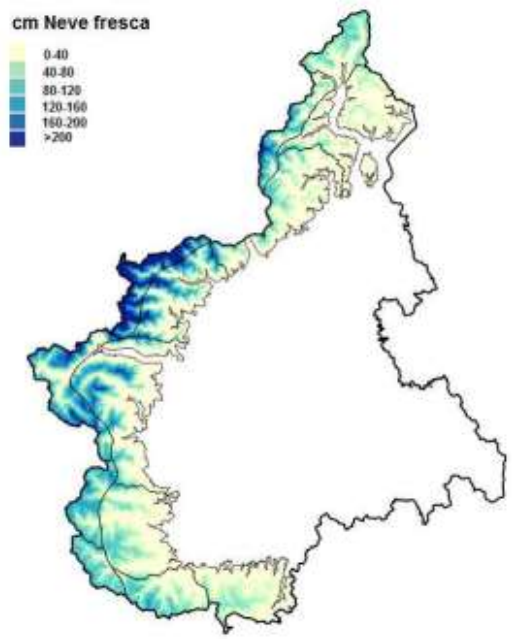
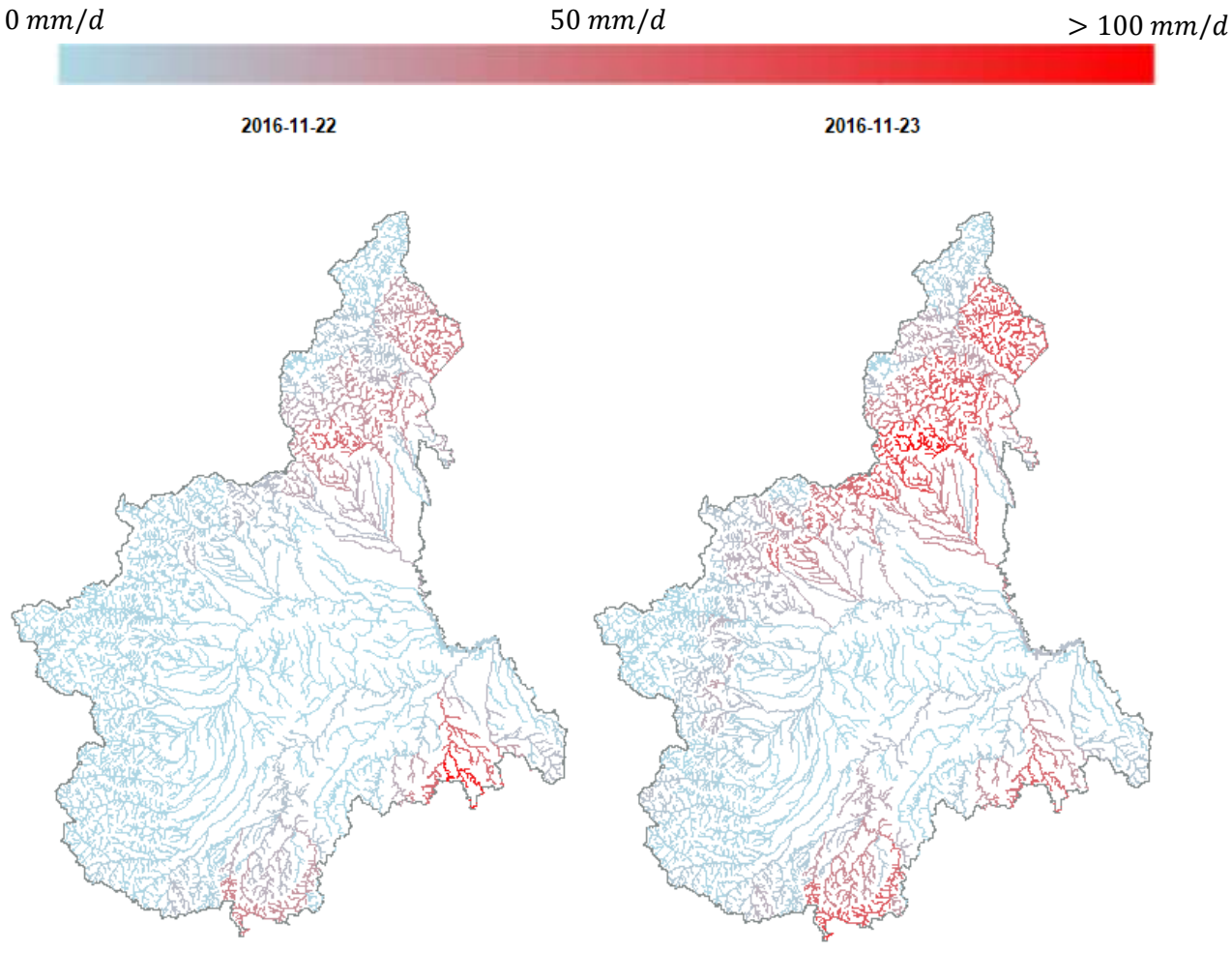


Figure 58 – Cumulative fresh snow over the entire event of November 2016 over the Piedmont region ("Gli eventi alluvionali in Piemonte – Evento del 21-25 novembre 2016", ARPA Piemonte, 2018)

Considering instead the response of the river network the event produced discharges of extreme entity along the Tanaro River and more in general in all the rivers in the west basins and in the Po River till the city of Turin. The uppermost section of the Tanaro River experienced the first precipitations already on the 21st, but the real flood generated later the 24th with the increase of the intensity of rainfalls; most of the values recorded represented the new observed maximum, overcoming even the observation of 1994. The flood of the Po River has been generated by the superposition, nearly in phase, of the waves coming from the Pellice and Po Rivers at their confluence, forming a particularly severe discharge. After the city of Turin, additional water coming from the left tributaries (that individually did not present exceptional discharges) reinforced the runoff that reached downstream San Sebastiano (*TO*). To synthesize the effects over the most affected areas, in Table 19 are reported the results of the statistical analysis for some indicative stations:

Bacino	Stazione	Portata al colmo [mc/sec]	TR [anni]	Q TR PGRA [mc/sec]		
				TR 20	TR 200	TR 500
SESIA	Sesia a Palestro	1800	<20	3700	5500	6300
DORA BALTEA	Dora Baltea a Verolengo	750	<20	1630	3000	3520
STURA DI LANZO	Stura di Lanzo a Torino	1500	20-200	1240	2080	2440
STURA DI LANZO	Stura di Lanzo a Lanzo	840	<20	1080	1810	2120
DORA RIPARIA	Dora Riparia a Torino	550	20-200	300	630	890
PELLICE	Chisone a San Martino	730	20-200	580	1100	1340
PELLICE	Pellice a Villafranca	1310	20-200	1010	1920	2340
PO	Chisola a La Loggia	440	200-500		400	450
PO	Sangone a Torino	430	<20	580	820	900
PO	Po a Villafranca	970	20	900	1300	1460
PO	Po a Carignano	2200	20-200	1600*	2400*	2720*
PO	Po a Moncalieri	2400	20-200	1750	2700	3100
PO	Po a Torino C.so Regina	2500	20-200	1730*	2600*	2950*
PO	Po a San Sebastiano	4430	20-200	3800 (1)	5900(1)	6600(1)
PO	Po a Crescentino	4990	<20	6100	-	10900
PO	Po a Casale Monferrato	4950	<20	6000	-	10000
PO	Po a Valenza	6120	<20	6500	7600*	10800
PO	Po a Isola S. Antonio	9950	20-500	8400	10300*	13300
TANARO	Bormida di Millesimo a Murialdo	480	500 (2)	-	-	-
TANARO	Bormida di Millesimo a Camerana	940	200 (2)	-	-	-
TANARO	Bormida di Millesimo a Cessole	1160	500 (2)	-	-	-
TANARO	Bormida di Spigno a Piana Crixia	740	20-200 (2)	-	-	-
TANARO	Bormida di Spigno a Mombaldone	1010	20-200 (2)	-	-	-
TANARO	Bormida a Cassine	2110	20-200	2010	3020	3400
TANARO	Tanaro a Garessio	830	200-500 (2)	-	-	-
TANARO	Tanaro a Piantorre	1430	200 (2)	-	-	-
TANARO	Tanaro a Farigliano	3120	> 500	1550	2300	2600
TANARO	Tanaro ad Alba	3440	500	2050	3050	3400
TANARO	Tanaro ad Asti	3450	20-200	2650	3550	3900
TANARO	Tanaro ad Alessandria	3470	200	2850	3400	4300
TANARO	Tanaro a Montecastello	3720	20-200	3250	5400	-

Table 19 –Peak discharges and return periods for some significant stations relative to the flood event of November 2016 ("Gli eventi alluvionali in Piemonte – Evento del 21-25 novembre 2016", ARPA Piemonte, 2018)



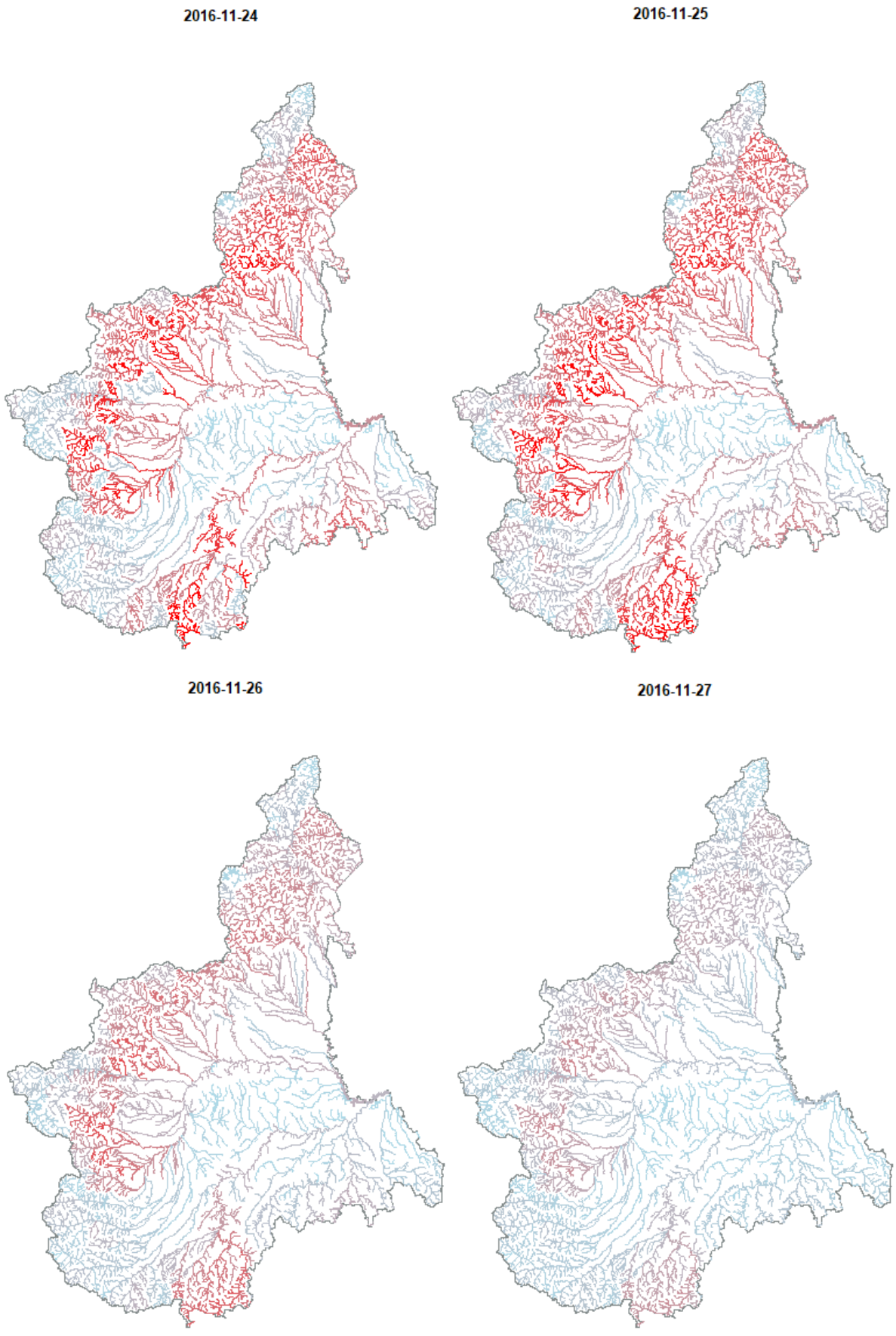


Figure 59 - Regional simulation (TUWmodel) using the HydroPASS regionalized parameters of the flood event in 2016

As for the previous events, the regionalized model reconstruction agrees with the description of the event as presented in the official reports. On the 22nd and 23rd of November the precipitation (and so the generated discharge) concentrated in the north and south sectors of the region to then extend to the western sector only afterward with peaks on the 24th of the month. A possible comparison for the fresh snow between the cumulative values in Figure 58 and the *SWE* produced by *TUWmodel* could be possible, but it goes beyond the scope of this study.

Flood 2020

The flooding event of the year 2020 has been a fast one, that started and concluded in a timeframe of just 48 hours. On the 1st of October was given a first warning (orange, following a traffic light coloring) for the following day, confirmed the following morning with a further forecasted intensification of the precipitations. The most intense rainfalls concentrated on the 2nd of October, interesting mainly the Biella, Vercelli, and Verbania municipalities and the surrounding areas (in the north) and the high Tanaro Valley (in the south, close to the border with the Liguria region). On the 3rd of October further intense precipitation kept interesting the Verbania areas.

During the entire duration of the event strong winds were recorded, with gusts over $120\ km/h$. The peaks precipitations were recorded by the rain gauges of the monitoring network (cumulated from the 1st to the 4th of October) in the localities of Valstrona (more than $650\ mm$), Mergozzo (over $600\ mm$) in the Verbania area, and in the localities of Limone Piemonte (close to $600\ mm$) and Garessio (more than $400\ mm$) in the high Tanaro Valley. Overall, the observed rainfalls overcame in these two days alone the 50% of the mean annual regional precipitations and total precipitation (over the region) on the 2nd of October represented the highest value since 1958 (when the historical series started).

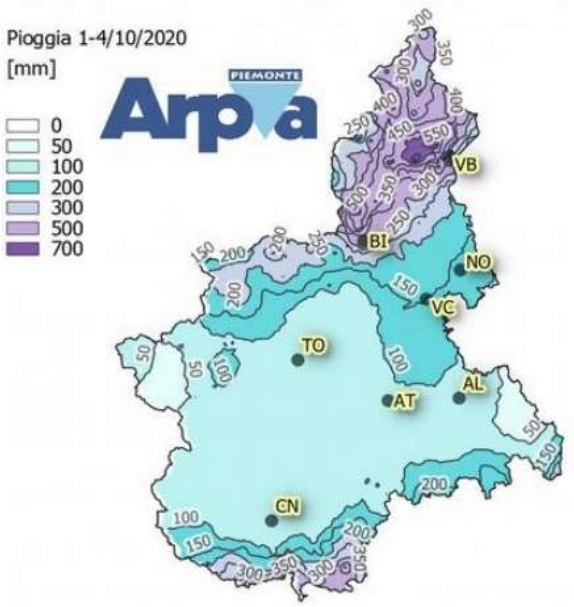


Figure 60 – Cumulative rainfall (from the 1st to the 4th of October 2020) over the Piedmont region ("Evento Alluvionale 2-3 Ottobre 2020", Regione Piemonte, 2020)

The analysis after the event showed how the data recorded by most of the rain gauges in the involved areas represented an exceedance of the reference expected values of precipitation (for all rainfall durations) with a return period of 200 years, further underlying the exceptionality of the event. As reference is reported in Figure 61 the example for the Limone Pancani station:

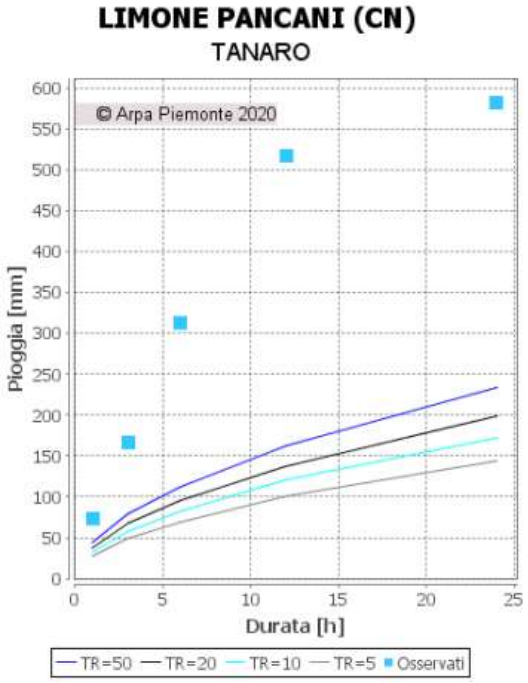
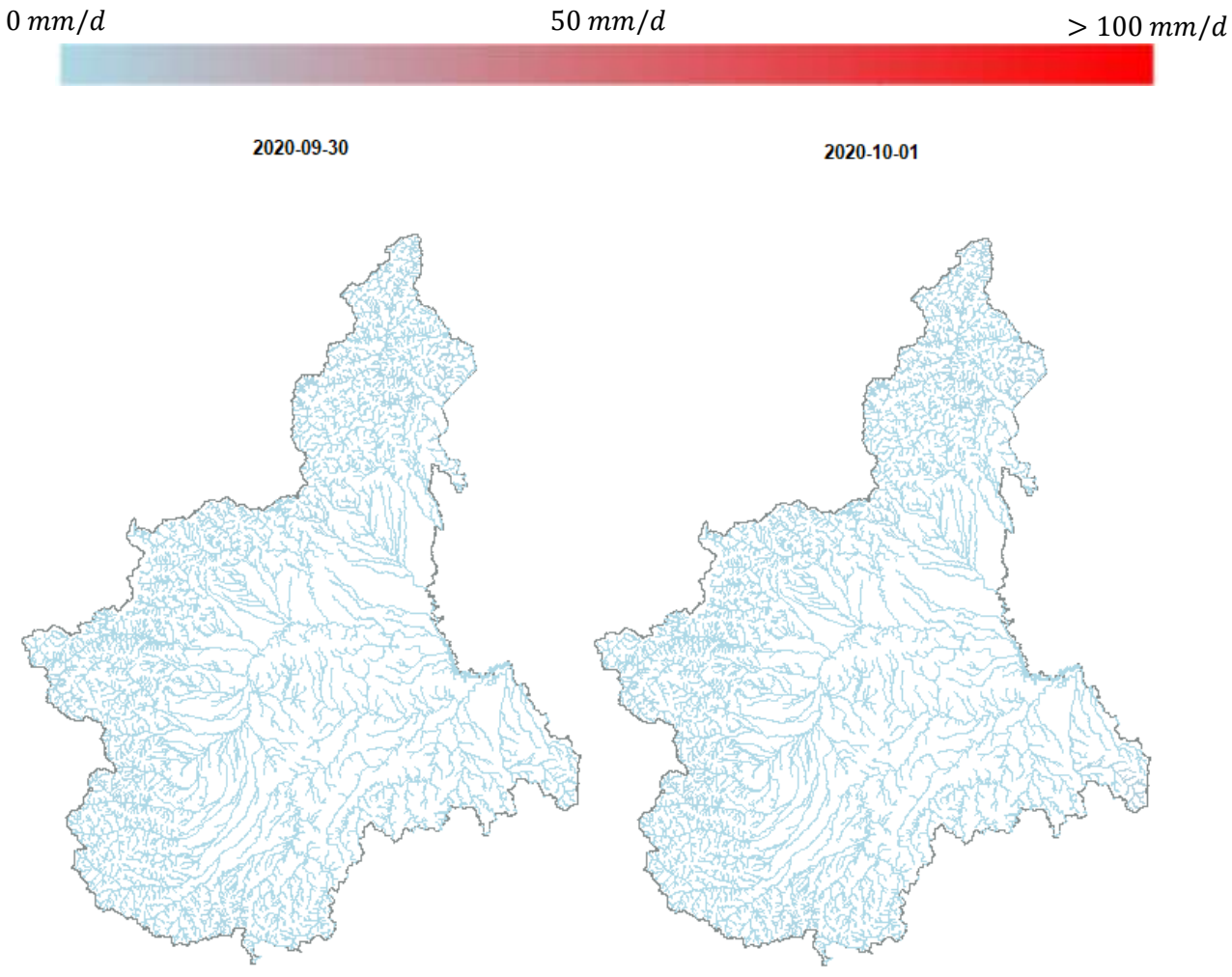


Figure 61 – Comparison between the reference values (for different duration and return periods) of precipitation and the observed rainfalls for the Limone Pancani station during the event of October 2020 ("Evento Alluvionale 2-3 Ottobre 2020", Regione Piemonte, 2020)

The precipitations transformed into discharge in the rivers, causing flooding (of both the primary and the secondary network) and diffused damages all over the struck areas. In the Tanaro Valley the Tanaro River and its tributaries (Vermenagna and Corsaglia Torrents) overflowed: at many locations the water level overcame the danger level and in some of them (like Ponte Nava and Garessio) the recorded values were higher than the data registered during the floods of 2016 presented before. In the municipality of Verbania the main issues were related to the Toce River with overflows in the localities of Mergozzo, Ornavasso, Premosello, Chiovenda and Pieve Vergonte. In the municipality of Biella similar problems were caused by the Cervo, Elva, Strona and Sessera Torrents. Probably the most intense discharges were observed in the Sesia River in the municipality of Vercelli: the recorded values for the river (and for the Toce Torrent) were higher than the reference for a return period of 200 years (closer to 500 years); here some of the most severe damages were recorded (collapse of bridges and destruction of embankments). Still in the province of Turin the Dora Baltea River overflowed and overcame the danger level.



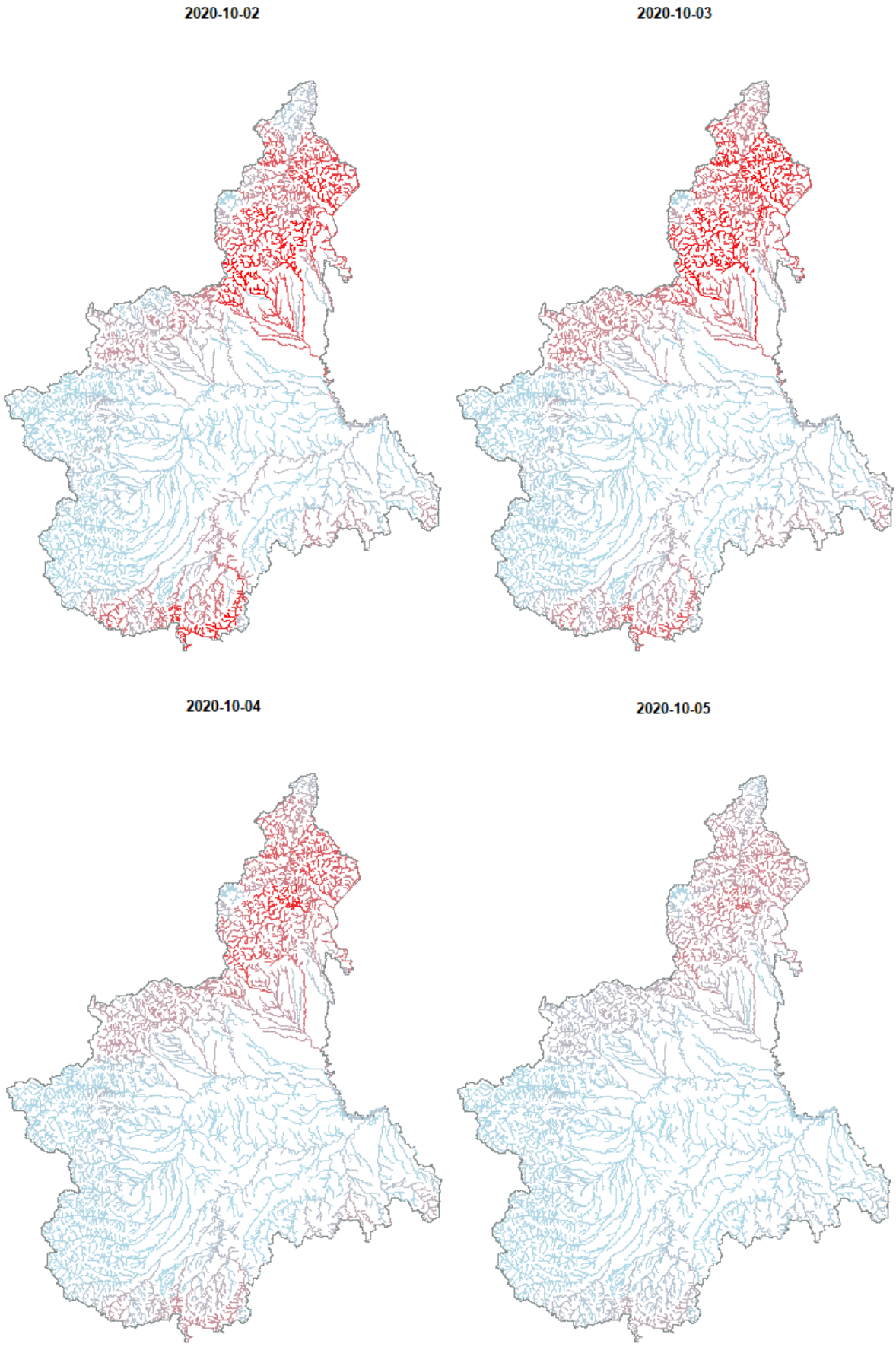


Figure 62 - Regional simulation (TUWmodel) using the HydroPASS regionalized parameters of the flood event in 2020

Comparing the simulated maps in Figure 62 with the distribution of cumulated precipitation in Figure 60, it can be noticed the very good match between the two. Not only the areas involved have been precisely reconstructed, but also the very fast timing of the event is well represented: in the first two days before the event no precipitation was recorded, and the discharges quickly reduced in the two following days.

8. CONCLUSIONS

The main goal of the study, as stated in the introduction, can be considered achieved, since the application of the regionalization procedure, through the *HydroPASS* algorithm, has been successful with results comparable with the ones obtained by Merz at al. in 2020. More in detail, starting from an average local lumped *ME* calibration of 0.86 (Figure 17), after regionalization a distributed *ME* of 0.70 has been reached (Table 15). Such results can be considered satisfactory and potentially useful for practical, meaningful applications. During the analysis of the results of the regionalization procedure also a reduced equifinality of the model parameters with respect to the uncertainty obtained after the local calibration has been observed (Figure 24); in fact, most of the *TUWmodel* parameters, with few exceptions, greatly reduced their spread around their mean values (Figure 42). This result, despite still underlying a non-complete understanding of the hydrological processes controlling runoff generation, suggests how a stronger relationship between specific model parameters sets and good *MEs* could, indeed, exist. Moreover, the distributed model parameters for the Piedmont region, over the cells of the reference grid, result spatially coherent with the morphology and topography of the territory, further confirming the consistency of the procedure. This aspect is particularly clear looking for example at one of the routing related parameters (like *croute* - Figure 46) that assumes distinct higher values along the entire Alpine sectors where the slopes are higher.

Nevertheless, some potential limits can still be underlined: they are mainly related to the local lumped parameter sets used for regionalization and to the data used for the calibration themselves, and only marginally to other aspects of the study. First, the observed discharges at the gauged locations were in many cases uncomplete (especially during the 90s), noisy or extremely short. Despite such problems made impossible the calibration for only 10 catchments out of the initial 127 available, they could have negatively influenced the *MEs* (for example see the influence of the number of years in Figure 18). Being the local lumped model efficiencies the starting point of the regionalization procedure, they may have indirectly negatively influenced of some percentual points also the final distributed *MEs*. Second, still related to the results of the lumped local calibrations, some calibrated catchments, mainly concentrated along the regional mountainous sectors (Figure 16) presented considerably lower efficiencies than the other basins: the introduction of snow cover observations as input data and of a snow related index inside the model efficiency (to include in the *ME* score the capability of *TUWmodel* in reconstructing the *SWE*) could reduce such effect, improving the average quality of the calibrations.

The third and final limit of the local calibration has been already underlined in chapter 4 in the Figures from 19 to 22. Despite a good ability of the model to reconstruct the catchments' hydrological signatures at the annual timescale, an apparent higher difficulty to do so at the seasonal timescale (during the summer-autumn months) has been instead encountered. Again, this aspect may be related to snow related processes and so the same fixes proposed for the previous issue may be helpful also in this case.

The second biggest limit of this study is the absence of a specific validation procedure: as explained in chapter 6, Pesce at al. reported good validation results proving the robustness of the *HydroPASS* algorithm, but we cannot properly state the same exact outcome would also apply for our case. A further evolution and continuation of this work should also include it, to further guarantee the spatial and temporal coherence of any following application. Nevertheless, the similarities between this study and the cited one in paragraph 6.4 are numerous (region, reference grid's resolution, gauged catchments used for the calibration), not invalidating completely the plausibility of our flood reconstruction application. The main issue could be the temporal coherence (they considered the period 2000-2020), but all the event simulated in this study are included or not distant in time (the furthest is in 1994) from the validation period they have considered. Last, another marginal issue is the non-coverage by the reference grid of the high Canton-Ticino Basin that may have negatively influenced the calibrations in that area.

On the other hand, many positive aspects, that opens to possible future implementations of this regionalization procedure are highlighted hereafter: first, the possibility of efficiently reconstruct discharges in the river network at ungauged locations. This is the main goal and application of the regional distributed parameters reported in this study, as shown in chapter 7 with the historical floods' reconstructions. Having a tool to not only reconstruct a full coherent spatial map of an event of which only partial observations are available, but potentially also reconstruct entire events with few or no observations (starting only from the climatological inputs), can be of great relevance in environmental studies. A clear example is the simulation for the flood of 1994 when many of the stations of the monitoring network failed, were destroyed during the event (in the most affected areas) or were not functioning at all in first place: the obtained simulated discharges and the distributed maps like the one presented in Figure 52 enables many possible considerations about what could realistically have happened. A better regional knowledge of the hydrological response, even if approximated, may also lead to a better understanding of the past criticalities on the territory, to a stronger awareness and bring to a better distribution and implementation of prevention and monitoring systems.

In second place, one of the most useful findings, purely under a practical standpoint, is the great flexibility of the *HydroPASS* algorithm itself and its partial independency from the catchment descriptors used to characterize the territory. Paragraph 6.4 shown how alternatively removing all the *CDs* belonging to one of the five considered groups and repeating the regionalization procedure led to nearly identical model efficiency *eCDF*. This fact directly translates into the possibility to reproduce similar procedures in regions with lower attributes data availability without excessively reduce the quality of the regionalization, making it suitable also for poorer or less mapped (in open databases) countries. Further verifications are still needed about the minimal number of catchment descriptors required (and in which possible combinations) to not degrade the quality of the regionalization; this can be easily performed with further *HydroPASS* runs checking the mutual possible combinations removing gradually more than one group of descriptors (but may be indeed a time-consuming operation).

A final proposed option, not considered inside this thesis, is the possibility to not stop at implementations focusing only on what happened in the past but also thinking about applications to future climate projections. From a practical point of view, once the regionalized model parameters have been obtained, nothing prevents their utilization with simulated future climate input to construct what would happen in possible futures (assuming the same regional functional relationship will still be valid). Such fictional climate inputs are for example generated inside climate models, following different possible scenarios, to try to understand what may lead in front of us and have been the center of an increasing interest in the scientific community in the last decades. The resolution used in this study (around $12\ km$ on the ground) is in fact compatible with the resolution of the climate input of a Regional Climate Model (*RCM*), that ranges around the tens of km, after the downscaling from a Global Climate Model (*GCM*). Having a regional hydrological model to be easily and directly coupled with a climate model can facilitate the production of ensembles to assess the uncertainty of such models in runoff generation inside climate change studies. Further testing and mutual integrations with climate models will be needed to validate such possibility and it opens to abundant future studies involving the innovative regionalization approach here described.

MAIN R SCRIPTS

```
# CATCHMENTS LOCAL LUMPED CALIBRATION #

# ------------------------------ INITIALIZATION ------------------------------ #
#Packages loading
library(hydroGOF)       #For KGE function
library(zoo)            #For timeseries data management (zoo objects)
library(TUWmodel)       #For TUWmodel function
library(DEoptim)        #For Differential Evolution algorithm

#Definition of the objective function
ME <- function(param, prec, airt, ep, area, disc) {
  simu <- TUWmodel(prec=prec, airt=airt, ep=ep, area=area, param)$q
  simu[is.nan(simu)] <- -999
  simu <- simu[-c(1:303)]  #Remove the warming period from simulated series
  obse <- disc[-c(1:303)]  #Remove the warming period from observed series
  kgeQ <- KGE(simu, obse, na.rm=TRUE) # Calculation of the KGE
  return(-kgeQ)              #Negative KGE (DEoptim minimize the function)
}


# ------------------------------ DATA LOADING ------------------------------ #
#Observed timeseries for all catchments (N series) [m^3/s]
Discharge <- load("Discharge.RData")

#"Train topology" for all catchments (N matrices)
train.topology <- load("Weights.RData")

#Additional information of the catchments
char_197catch <- load("char_197catch.RData")

#Precipitation timeseries for pixels of the grid (matrix 23276*500)[mm/d]
prec <- load("Prec.RData")

#Mean temperature timeseries for pixels of the grid (matrix 23276*500) [°C]
Tmean <- load("Tmean.RData")

#PET timeseries for pixels of the grid (matrix 23276*500) [mm/d]
pet <- load("Pet.RData")


# ---------------------- DATA WINDOWING (60 years of data) ------------------- #
prec <- window(prec, start=as.Date('1961-01-01'), end=as.Date('2020-12-31'))
tmean <- window(tmean, start=as.Date('1961-01-01'), end=as.Date('2020-12-31'))
pet <- window(pet, start=as.Date('1961-01-01'), end=as.Date('2020-12-31'))


# ------------------------------ CALIBRATION ------------------------------ #
NOMI <- names(Discharge)              #Vector with all the names of the catchments
calibr <- vector('list', length(NOMI))#List to save the results
names(calibr) <- NOMI                 #Renamed the elements of the list


for(i in 1:127){
  #Extraction observed discharge for the current catchment
```

```
  Qobs <- Discharge[[NOMI[i]]]


#Fill the observed discharge series with NA at the beginning and at the end to
match the length of the

  #Climate variables observation series
  dummy0 <- zoo(NA, seq(as.Date('1900-01-01'), as.Date('2021-01-01'), by='day'))
  Qobs <- window(merge(Qobs, dummy0)[,1], start=min(time(tmean)),
end=max(time(tmean)))


  #Calculation of the conversion factor between m^3/s and mm/d
  m3s2mmd_2 <- char_197catch[char_197catch$codice == NOMI[i],
'area_bacinokm']/(3.6*24)

  #Extraction (as numbers) of the pixel ID (1 to 500)
  temp <- as.numeric(colnames(prec))

  #Extraction only of the pixels touched by the catchment
  qualiPixels <- train.topology[[NOMI[i]]][,1]

  #Determination of the position inside temp of the pixels
  temp2 <- vector()
  for (j in 1:length(qualiPixels)){
    temp2[j] <- which(qualiPixels[j] == temp)
  }

  #Extraction of precipitation only for the needed pixels
  precCat <- round(prec[, temp2, drop=FALSE],3)

  #Extraction of mean temperature only for the needed pixels
  tmeanCat <- round(tmean[, temp2, drop=FALSE],3)

  #Extraction of evapotranspiration only for the needed pixels
  petCat <- pet[, temp2, drop=FALSE]

  #Vector with the weight of each needed pixel
  weightsCat <- round(train.topology[[NOMI[i]]]$effarea,2)

  #Renaming of the column of new filtered matrices
  names(precCat) <- train.topology[[NOMI[i]]]$grd
  names(tmeanCat) <- train.topology[[NOMI[i]]]$grd
  names(petCat) <- train.topology[[NOMI[i]]]$grd
  names(weightsCat) <- train.topology[[NOMI[i]]]$grd

  # Calibration with DEoptim
  timestamp()
  calibr[[NOMI[i]]] <- try(DEoptim(fn=ME, lower=c(SCF=0.9, DDF=0.0, Tr=1.0,
  Ts=-3.0, Tm=-2.0, LPrat=0.0, FC=0.0,   BETA=0.0,  k0=0.0, k1=2.0,  k2=30.0,
  lsuz=1.0, cperc=0.0, bmax=0.0, croute=0.0),
  upper=c(SCF=1.5, DDF=5.0, Tr=3.0, Ts=1.0,  Tm=2.0, LPrat=1.0, FC=600.0,
  BETA=20.0, k0=2.0, k1=30.0, k2=250.0, lsuz=100.0, cperc=8.0, bmax=30.0,
  croute=50.0),
  control=DEoptim.control(NP=NA, itermax=200, reltol=1e-3, steptol=10, trace=5,
  parallelType=1, packages=c('TUWmodel','hydroGOF')),
  prec=precCat, airt=tmeanCat, ep=petCat, area=weightsCat,
  disc=as.numeric(Qobs)/m3s2mmd_2))
  timestamp()
```

```r
# ---------------------------------- SAVING ------------------------------------ #
save(calibr, file='Calibration.RData', compress='xz')
                              #HydroPASS APPLICATION#

# ---------------------- INITIALIZATION AND DATA LOADING ---------------------- #

library(TUWmodel)
library(hydroGOF)
library(zoo)
library(rpart)

load('data117cat305pxl.RData')
#It contains 30 lumped calibrated parameter sets for each catchment, descriptors
#for each catchment and each pixel, the "train topology", the observed discharges
#for each basin and the climate variables for each pixel


# ------------------------------ PASS FUNCTION -------------------------------- #
#Y is a list (N.cat) of data.frames (XXX x N.par) with locally lumped calibrated
#model parameters OR previously obtained PASS output.

#X.cat is a matrix or data.frame (N.cat x N.dsc) of catchment descriptors

#X.grd is a matrix or data.frame (N.grd x N.dsc) of model unit/pixel descriptors

#grd2cat is a list (N.cat) of model unit/pixel names belonging to each catchment

#model.edd.fn is the function to be optimized (maximized). The function should
#have as its first argument the vector/matrix of real-valued parameters to
#optimize, as second argument the catchment index, and return a scalar real result
#with maximum equal to 1. 'NA' and 'NaN' values are not allowed.

#lower and upper are two vectors specifying scalar real lower and upper bounds on
#each parameter to be optimized, so that the i-th element of 'lower' and 'upper'
#applies to the i-th

PASS <- function (Y, X.cat, X.grd, grd2cat, model.eff.fn, lower, upper,
                  options=PASS.options(), ...) {

     #Select parameter sets
  N.cat <- nrow(X.cat)  # Number of catchments
  N.grd <- nrow(X.grd)
  if (class(Y) != 'PASS') {
    train.parameters <- Y
    options <- do.call(PASS.options, as.list(options))
    overall.eff.threshold <- rep(-999, options$nGroups)
    PASS.group <- list()
    PASS.save <- list()
  } else {
    previousPASSout <- Y
    overall.eff.threshold <- sapply(previousPASSout$groups, function (x)
x$overall.eff)
    options <- do.call(PASS.options, as.list(options))
    options$nGroups <- length(previousPASSout$groups)
    train.parameters <- previousPASSout$train.parameters.updated
    PASS.group <- list()
    PASS.save <- previousPASSout
  }
```

```
      #Parameter normalization (to calculate distances in the regional consistency
      #algorithm)
  train.parameters.norm <- train.parameters
  for (i.cat in 1:N.cat) {
    dummy_max <- t(upper %*% t(rep(1, nrow(train.parameters[[i.cat]]))))
    dummy_min <- t(lower %*% t(rep(1, nrow(train.parameters[[i.cat]]))))
    train.parameters.norm[[i.cat]][,-1] <- (train.parameters[[i.cat]][,-1] -
                                            dummy_min)/(dummy_max - dummy_min)
      #Transformation (the first column contains efficiencies)
  }
  dummy_max.cat <- t(upper %*% t(rep(1, N.cat)))
  dummy_min.cat <- t(lower %*% t(rep(1, N.cat)))
  dummy_max.grd <- t(upper %*% t(rep(1, N.grd)))
  dummy_min.grd <- t(lower %*% t(rep(1, N.grd)))

  cat('%%%% ------------- START ------------- %%%%\n')
  time0 <- Sys.time()
  for (i.PASS in 1:options$maxLoops) {
    if (i.PASS %% round(options$maxLoops/10) == 0) cat('     Loop', i.PASS, 'out
of', options$maxLoops, 'loops\n')
    i.Group <- sample(options$nGroups, 1)
    if (options$sampling == 'random') {
      selected.parameters <- t(sapply(train.parameters, function (x)
                                      x[sample(nrow(x), 1),]))
      #Matrix with N rows (n. of catchments) and M columns (n. of parameters)
    } else if (options$sampling == 'optim') {
      selected.parameters <- previousPASSout$groups[[i.Group]]$selected.parameters
      select.cat <- sample(N.cat, round(N.cat*options$optim.subset.cat))
      selected.parameters[select.cat,] <- t(sapply(train.parameters[select.cat],
function (x) x[sample(nrow(x), 1),]))
    }

      #Parameter normalization (to calculate distances in the regional consistency
      algorithm)
    selected.parameters[,-1] <- (selected.parameters[,-1] -
      dummy_min.cat)/(dummy_max.cat - dummy_min.cat)  #Transformation

      #Regional consistency algorithm:
    for (i.REG in 1:options$REGloops) {
      regionalization <- .DT.app(par.in=selected.parameters[, -1],
catch_CDs=X.cat, grds_CDs=X.grd)  # output = cat.par.pred and grd.par.pred
      for (i.cat in 1:N.cat) {
        par.reg.cat <- regionalization$cat.par.pred[i.cat,]
        distances <- (train.parameters.norm[[i.cat]][,-1] - t(par.reg.cat %*%
t(rep(1, nrow(train.parameters.norm[[i.cat]])))))^2
        selected.parameters[i.cat,] <-
train.parameters.norm[[i.cat]][which.min(apply(distances, 1, sum, na.rm=TRUE)),]
      }
    }
    colnames(regionalization$cat.par.pred) <- colnames(selected.parameters)[-1]
    rownames(regionalization$cat.par.pred) <- rownames(X.cat)
    colnames(regionalization$grd.par.pred) <- colnames(selected.parameters)[-1]
    rownames(regionalization$grd.par.pred) <- rownames(X.grd)
    regionalization$cat.par.pred[regionalization$cat.par.pred > 1] <- 1
    regionalization$cat.par.pred[regionalization$cat.par.pred < 0] <- 0
    regionalization$grd.par.pred[regionalization$grd.par.pred > 1] <- 1
    regionalization$grd.par.pred[regionalization$grd.par.pred < 0] <- 0
    regionalization$cat.par.pred <- regionalization$cat.par.pred*(dummy_max.cat -
dummy_min.cat) + dummy_min.cat
    regionalization$grd.par.pred <- regionalization$grd.par.pred*(dummy_max.grd -
dummy_min.grd) + dummy_min.grd
```

```
 eff.lump <- rep(NA, N.cat)
    eff.dist <- rep(NA, N.cat)
    for (i.cat in 1:N.cat) {
      #Lumped
      param.cat <- regionalization$cat.par.pred[i.cat, ]
      param.cat[is.na(param.cat)] <- -999
      eff.lump[i.cat] <- model.eff.fn(param.cat, i.cat, ...)

      if (eff.lump[i.cat] >
options$proportion.max.eff.update*max(train.parameters[[i.cat]][, 1])) {     # ME
        train.parameters[[i.cat]] <- rbind(train.parameters[[i.cat]],
c(eff.lump[i.cat], param.cat))
      }

      #Distributed
      param.grd <- regionalization$grd.par.pred[as.character(grd2cat[[i.cat]]), ]
      param.grd[is.na(param.grd)] <- -999
      eff.dist[i.cat] <- model.eff.fn(param.grd, i.cat, ...)
    }

    overall.eff.dist <- mean(eff.dist)

    if (options$sampling == 'random') {
      i.Group <- which.min(overall.eff.threshold)
    }
    if (overall.eff.dist > overall.eff.threshold[i.Group]) {
      overall.eff.threshold[i.Group] <- overall.eff.dist
      PASS.group[['overall.eff']] <- overall.eff.dist
      PASS.group[['selected.parameters']] <- selected.parameters
      PASS.group[['regionalized.parameters']] <- regionalization
      PASS.group[['cat.eff.lump']] <- eff.lump
      PASS.group[['cat.eff.dist']] <- eff.dist
      PASS.save[['groups']][[paste('Group', i.Group, sep='')]] <- PASS.group
    }
  }
  PASS.save[['train.parameters.updated']] <- train.parameters
  PASS.save[['PASS.options']] <- options
  cat('%%%% -------------  END  ------------- %%%%\n')
  time1 <- Sys.time()
  print(time1 - time0)
  class(PASS.save) <- 'PASS'
  return(PASS.save)
}


# --------------------------- ADDITIONAL PASS FUNCTIONS ----------------------- #
print.PASS <- function (x, ...) {
  cat('Output of the PArameter Set Shuffling algorithm:\n\n')
  cat(' number of catchments:',
nrow(x$groups[[1]]$regionalized.parameters$cat.par.pred), '\n')
  cat(' number of model units (e.g. pixels):',
nrow(x$groups[[1]]$regionalized.parameters$grd.par.pred), '\n')
  cat('sampling strategy:', x$PASS.options$sampling, '\n')
  cat(' number of loops:', x$PASS.options$maxLoops, '\n')
  cat(' number of groups:', x$PASS.options$nGroups, '\n')
  cat(' number of loops for regional consistency:', x$PASS.options$REGloops, '\n')
  if (x$PASS.options$sampling == 'optim') {
    cat(' proportion of randomized parameters (when optim):',
x$PASS.options$optim.subset.cat, '\n')
  }
  cat('lumped regionalization efficiencies:', '\n')
```

```
  print(round(apply(sapply(x$groups, function(x) x$cat.eff.lump), 2, summary), 3))
 cat('distributed regionalization efficiencies:', '\n')
  print(round(apply(sapply(x$groups, function(x) x$cat.eff.dist), 2, summary), 3))
  cat(' updated number of train parameter sets:',
sum(sapply(x$train.parameters.updated, nrow)), '\n')
}


PASS.options <- function(maxLoops=100, nGroups=10, REGloops=5,
                         generalized.mean.power=-1,
proportion.max.eff.update=0.95,
                         sampling='random', optim.subset.cat=0.7) {
  list(maxLoops=maxLoops, nGroups=nGroups, REGloops=REGloops,
       generalized.mean.power=generalized.mean.power,
proportion.max.eff.update=proportion.max.eff.update,
       sampling=sampling, optim.subset.cat=optim.subset.cat)
}


# ------------------------------ RANDOM FOREST ROUTINE ------------------------- #
.DT.app <- function(par.in, catch_CDs, grds_CDs)  {
  iparam <- ncol(par.in)

  catch_CDs <- as.data.frame(catch_CDs)
  grds_CDs <- as.data.frame(grds_CDs)
  DT.save <- list()
  par.pred <- array(NA, dim=c(nrow(catch_CDs), iparam))
  grd.par.pred <- array(NA, dim=c(nrow(grds_CDs), iparam))

  ###Account for NAs in descriptors
  grds_CDs.temp <- grds_CDs
  goodgrds <- complete.cases(grds_CDs.temp)
  grds_CDs.temp[is.na(grds_CDs.temp)] <- -9999.99
  par.in.temp <- as.data.frame(par.in)

  for (ipar in 1:iparam) {
    trainX <- cbind(par.in.temp[,ipar], catch_CDs)
    colnames(trainX) <- c("Par", colnames(catch_CDs))
    DT.pred <- rpart(Par ~ ., data=trainX,
                     control=rpart.control(minsplit=min(20, nrow(trainX)/2)))
    DT.save[[ipar]] <- DT.pred
    #Catchments
    par.pred[,ipar] <- predict(DT.pred, catch_CDs)
    #Grid pixels
    grd.par.pred[,ipar] <- predict(DT.pred, grds_CDs.temp)
  }
  grd.par.pred[!goodgrds,] <- NA
  DT.app.back <- list()
  DT.app.back$cat.par.pred <- par.pred
  DT.app.back$grd.par.pred <- grd.par.pred
  return(DT.app.back)
}
```

```
# ------------------------------ OBJECTIVE FUNCITON ---------------------------- #
ME.TUWmodel <- function(param, cat.number, grdname, prec, airt, ep, area, disc,
iwarmup=303) {
  GRDNAME <- grdname[[cat.number]]
  AREA <- area[[cat.number]]
  PREC <- prec[, as.character(GRDNAME)]
  AIRT <- airt[, as.character(GRDNAME)]
  EP <- ep[, as.character(GRDNAME)]
  DISC <- disc[, cat.number]
  if (!is.null(dim(param))) param <- t(param)
  if(length(AREA) != 1){
    simu <- TUWmodel(prec=PREC, airt=AIRT, ep=EP,
                     area=AREA, param=param)
    simu <- simu$q[-c(1:iwarmup)]                    #Remove the warming period
  }
  if(length(AREA) == 1){
    simu <- TUWmodel(prec=as.numeric(PREC), airt=as.numeric(AIRT),
ep=as.numeric(EP),
                     area=AREA, param=param)
    simu <- t(simu$q)[-c(1:iwarmup)]                 #Remove the warming period
  }
  simu[is.na(simu)] <- -999
  obse <- DISC[-c(1:iwarmup)]                        #Remove the warming period
  me <- KGE(simu, obse, na.rm=TRUE)
  if (is.na(me)) me <- -999
  return(me)                                         #KGE calculation
}


# ------------------------------ START APPLICATION ---------------------------- #
#Select 30 best sets of parameters for each catchment that satisfies the selection
#condition
train.parameters <-
data117cat305pxl$local_param.calibraNOneve[sapply(data117cat305pxl$local_param.cal
ibraNOneve, function(x) max(x[,1])) > 0.8]  # 90

#Extraction of descriptors for catchments and grid + "train topology"
cat.CD <- data117cat305pxl$cat.CD[rownames(data117cat305pxl$cat.CD) %in%
names(train.parameters), ]
grd.CD <-data117cat305pxl$grd.CD
topology <- data117cat305pxl$grd2cat[names(train.parameters)]

#Extraction observed discharges
qobs <- data117cat305pxl$qobs[, names(train.parameters)]


# ---------------------- 1000 ITERATIONS IN "RANDOM" MODE --------------------- #
timestamp()
run01 <- PASS(Y=train.parameters,
              X.cat=cat.CD,
              X.grd=grd.CD,
              grd2cat=sapply(topology, function(x) x$grd.name),
              model.eff.fn=ME.TUWmodel,
              lower=c(SCF=0.9, DDF=0.0, Tr=1.01, Ts=-3.0, Tm=-2.0,
                      LPrat=0.0, FC=0.0, BETA=0.0, k0=0.0, k1=2.0, k2=30.0,
                      lsuz=1.0, cperc=0.0, bmax=0.0, croute=0.0),
              upper=c(SCF=1.5, DDF=5.0, Tr=3.0, Ts=1.0, Tm=2.0,
                      LPrat=1.0, FC=600.0, BETA=20.0, k0=2.0, k1=30.0, k2=250.0,
                      lsuz=100.0, cperc=8.0, bmax=30.0, croute=50.0),
              options=PASS.options(maxLoops=100, nGroups=10, REGloops=5,
sampling='random'),
```

```
              prec=data117cat305pxl$prec,
              airt=data117cat305pxl$tmean,
              ep=data117cat305pxl$pet,
              area=sapply(topology, function(x) x$grd.weightCat),
              grdname=sapply(topology, function(x) x$grd.name),
              disc=qobs)
print(run01)
save(run01, file='run01_TUW.RData', compress='xz')
timestamp()

timestamp()
for (rr in 2:10) {
  run01 <- PASS(Y=run01,
                X.cat=cat.CD,
                X.grd=grd.CD,
                grd2cat=sapply(topology, function(x) x$grd.name),
                model.eff.fn=ME.TUWmodel,
                lower=c(SCF=0.9, DDF=0.0, Tr=1.01, Ts=-3.0, Tm=-2.0,
                        LPrat=0.0, FC=0.0, BETA=0.0, k0=0.0, k1=2.0, k2=30.0,
                        lsuz=1.0, cperc=0.0, bmax=0.0, croute=0.0),
                upper=c(SCF=1.5, DDF=5.0, Tr=3.0, Ts=1.0, Tm=2.0,
                        LPrat=1.0, FC=600.0, BETA=20.0, k0=2.0, k1=30.0, k2=250.0,
                        lsuz=100.0, cperc=8.0, bmax=30.0, croute=50.0),
                options=PASS.options(maxLoops=100, nGroups=10, REGloops=5,
sampling='random'),
                prec=data117cat305pxl$prec,
                airt=data117cat305pxl$tmean,
                ep=data117cat305pxl$pet,
                area=sapply(topology, function(x) x$grd.weightCat),
                grdname=sapply(topology, function(x) x$grd.name),
                disc=qobs)
  print(run01)
  save(run01, file='run01_TUW.RData', compress='xz')
}
timestamp()


# ---------------------- 1000 ITERATIONS IN "OPTIM" MODE ---------------------- #
timestamp()
run02 <- PASS(Y=run01,
              X.cat=cat.CD,
              X.grd=grd.CD,
              grd2cat=sapply(topology, function(x) x$grd.name),
              model.eff.fn=ME.TUWmodel,
              lower=c(SCF=0.9, DDF=0.0, Tr=1.01, Ts=-3.0, Tm=-2.0,
                      LPrat=0.0, FC=0.0, BETA=0.0, k0=0.0, k1=2.0, k2=30.0,
                      lsuz=1.0, cperc=0.0, bmax=0.0, croute=0.0),
              upper=c(SCF=1.5, DDF=5.0, Tr=3.0, Ts=1.0, Tm=2.0,
                      LPrat=1.0, FC=600.0, BETA=20.0, k0=2.0, k1=30.0, k2=250.0,
                      lsuz=100.0, cperc=8.0, bmax=30.0, croute=50.0),
              options=PASS.options(maxLoops=100, nGroups=10, REGloops=4,
sampling='optim', optim.subset.cat=0.7),
              prec=data117cat305pxl$prec,
              airt=data117cat305pxl$tmean,
              ep=data117cat305pxl$pet,
              area=sapply(topology, function(x) x$grd.weightCat),
              grdname=sapply(topology, function(x) x$grd.name),
              disc=qobs)
print(run02)
save(run02, file='run02_TUW.RData', compress='xz')
```

```
timestamp()

timestamp()
for (rr in 2:10) {
  run02 <- PASS(Y=run02,
                X.cat=cat.CD,
                X.grd=grd.CD,
                grd2cat=sapply(topology, function(x) x$grd.name),
                model.eff.fn=ME.TUWmodel,
                lower=c(SCF=0.9, DDF=0.0, Tr=1.01, Ts=-3.0, Tm=-2.0,
                        LPrat=0.0, FC=0.0, BETA=0.0, k0=0.0, k1=2.0, k2=30.0,
                        lsuz=1.0, cperc=0.0, bmax=0.0, croute=0.0),
                upper=c(SCF=1.5, DDF=5.0, Tr=3.0, Ts=1.0, Tm=2.0,
                        LPrat=1.0, FC=600.0, BETA=20.0, k0=2.0, k1=30.0, k2=250.0,
                        lsuz=100.0, cperc=8.0, bmax=30.0, croute=50.0),
                options=PASS.options(maxLoops=100, nGroups=10, REGloops=4,
sampling='optim', optim.subset.cat=0.7),
                prec=data117cat305pxl$prec,
                airt=data117cat305pxl$tmean,
                ep=data117cat305pxl$pet,
                area=sapply(topology, function(x) x$grd.weightCat),
                grdname=sapply(topology, function(x) x$grd.name),
                disc=qobs)
  print(run02)
  save(run02, file='run02_TUW.RData', compress='xz')
}
timestamp()
```

!!! All the functions reported in the head of this script can now be found inside the *R* package HydroPASS downloadable from CRAN !!!

```
                        # REGIONAL FLOODS RECONSTRUCTION #

# ------------------------------- INITIALIZATION --------------------------- #
#Packages loading
library(raster)
library(rgdal)
library(sf)
library(tidyverse)
library(zoo)
library(TUWmodel)
library(maptools)
library(scales)

#Data loading
load("run02_TUW.RData")                  #It contains regionalization PASS results
load("data117cat500pxl.RData")           #It contains the climate timeseries
bacini <- st_read("C_ZYHD_Italia.shp")   #Load JRC basins shapefile
rivers <- st_read("c_tr_Italia.shp")     #Load JRC rivers shapefile
Piemonte <- st_read("Ambiti_amministrativi-Regione.shp") #Load Piedmont borders
grid <- st_read("griglia.shp")           #Load reference grid in UTM32


# --------------- INTERSECTION BETWEEN REGIONAL BORDDERS AND BASINS ------------ #
crs(bacini)      #Check the reference system of the JRC basins --> Lambert
bacini <- st_transform(bacini,crs = crs(grid))    #Basins in UTM32
rivers <- st_transform(rivers,crs = crs(grid))    #Rivers in UTM32
Piemonte <- Piemonte[c(11,12)]                    #Keeping only name and Geometry
grid[,1] <- c(1:500)                   #Renaming of the grid cells (from 1 to 500)
area_grid <- st_area(grid)                        #Evaluation area of each grid cell
colnames(grid)[1] <- "Pxl"                        #Renaming column of the cells
bacini$area_b <- st_area(bacini)                  #Calculation area of each JRC basin
names(bacini)[1] <- "ID"                          #Renomination column "ID"

#Intersection between regional borders and basins (638 remaining)
qualibacini <- st_intersection(bacini,Piemonte)


# ---------------------- EVALUATION TEMPORARY "TRAIN TOPOLOGY" ------------------ #
#Calculation intersection JRC basins and reference grid
intersection <- st_intersection(grid, qualibacini)

#Calculation areas of each pixel intersection
intersection$area_Pxl <- as.numeric(st_area(intersection))

for (i in 1:dim(intersection)[1]){
  #Normalization of pixel area intersections
  intersection$area_Pxl[i] <-
  intersection$area_Pxl[i]/area_grid[intersection$Pxl[i]]
}

#Kept only basin "ID", numeration, normalized pixel areas and geometry
intersection <- intersection[,c(2,1,56)]
final <- intersection %>%
  as_tibble() %>%                          #Change format
  group_by(ID, Pxl) %>%
  summarize(area = sum(area_Pxl))

#Creation of a list for a temporary "train topology"
train.topology_b <- vector('list',dim(qualibacini)[1])
for (i in 1:length(train.topology_b)){
  cat(i)
```

```
  dummy <- filter(final,ID == qualibacini$ID[i])
  dummy <- as.data.frame(dummy)    #Calculation of the temporary "train topology"
  dummy <- dummy[,-1]
  colnames(dummy) <- c("grdname","effarea")
  train.topology_b[[i]] <- dummy
}
names(train.topology_b) <- qualibacini$ID       #Rename the element of the list


# -------------------- EXCLUDING POLYGONS EXTERNAL TO GRID --------------------- #
NOMI <- names(train.topology_b)                 #Vector with "ID" JRC basins
#Extraction of regionalized PASS parameters over the grid
param <-
run02[["groups"]][["Group5"]][["regionalized.parameters"]][["grd.par.pred"]]
control <- as.numeric(rownames(param))          #Numeration of the 305 used pixels

#Elimination of any basin outside of the 305 pixels (633)
for (i in 1:length(NOMI)){
  grdname <- as.numeric(train.topology_b[[NOMI[i]]]$grdname)
  if(sum(grdname %in% control) < length(grdname)){
    train.topology_b[[NOMI[i]]] <- NULL
  }
}

NOMI <- names(train.topology_b)                 #Update of the "ID" vector


# ----------------------------- EXTRACT RIVER NETWORK  ------------------------- #
#Removal of the geometry of the removed basins
qualibacini <- qualibacini[as.character(qualibacini$ID) %in% NOMI,]
NOMI <- qualibacini$ZHYD                                #Vector with ZHYD codes

#Removal of the river geometry of the removed basins
qualirivers <- rivers[rivers$ZHYD %in% NOMI,]


# ---------------------------- NEW CATCHMENTS GEOMETRY --------------------#
Final_CAT <- vector('list')             #Creation list for new catchments geometry

#Determination new catchments geometry (requires some time)
for (i in 1:dim(qualibacini)[1]) {
  CAT <- vector('list')
  print(i)
  timestamp()
  zhyd0 <- qualibacini$ZHYD[i]
  zhyds <- zhyd0
  lengthminus1=0
  while(length(zhyds) > lengthminus1) {
    lengthminus1 <- length(zhyds)
    zhyds <- unique(c(zhyd0, qualibacini$ZHYD[qualibacini$NextDown %in% zhyds]))
  }
  IDs <- c(NA,zhyd0)[1 + (qualibacini$ZHYD %in% zhyds)]
  dummy <- st_union(qualibacini$geometry[!is.na(IDs)])
  dummy2 <- qualirivers$geometry[qualirivers$ZHYD == zhyd0]
  CAT$BACINO <- dummy
  CAT$FIUMI <- dummy2
  Final_CAT[[i]] <- CAT
}
names(Final_CAT) <- qualibacini$ZHYD                    #Rename list
```

```
AREA <- vector()       #Vector for the Areal extension of the new basins' geometry
  for (i in 1:length(Final_CAT)){
#Calculation of the areas
    cat(i)
    timestamp()
    AREA[i] <- st_area(Final_CAT[[i]]$BACINO)
  }
  AREA <- AREA/1000000              #Area in Km^2


# ---------------- EXTRACTION TRAIN TOPOLOGY AGGREGATED CATCHMENTS ------------ #
#List for the new train.topology_b
train.topology_b <- vector('list',length(Final_CAT))
names(train.topology_b) <- NOMI                          #Rename list's elements

#Calculation new "train topology" (requires some minutes)
for (j in 1:length(train.topology_b)){
  cat(j)
  timestamp()
  intersection <- st_intersection(grid, Final_CAT[[j]]$BACINO)
  intersection$area_Pxl <- as.numeric(st_area(intersection))
  for (i in 1:dim(intersection)[1]){
    intersection$area_Pxl[i] <-
intersection$area_Pxl[i]/area_grid[intersection$Pxl[i]]
  }
  final <- intersection %>%
    as_tibble() %>%
    group_by(Pxl) %>%
    summarize(area = sum(area_Pxl))
  final <- as.data.frame(final)
  colnames(final) <- c("grdname","effarea")
  train.topology_b[[NOMI[j]]] <- final
}


# ---------------------- SIMULATE DISCHARGES OVER CATCHMENTS ------------------- #
#List for regional simulated discharges
SIMUL_PASS <- vector('list',length(train.topology_b))
names(SIMUL_PASS) <- NOMI                                #Rename list's elements
date <- index(data117cat500pxl$prec[,1])                 #Estraction vector of dates

#Discharge simulation with TUWmodel (it requires several minutes)
for (i in 1:length(NOMI)){
  cat(i)
  timestamp()
  grdname <- as.character(train.topology_b[[NOMI[i]]]$grdname)
  prec <- data117cat500pxl$prec[,grdname]
  tmean <- data117cat500pxl$tmean[,grdname]
  pet <- data117cat500pxl$pet[,grdname]
  area <- train.topology_b[[NOMI[i]]]$effarea
  param.grd <- param[grdname,]
  if (!is.null(dim(param.grd))) param.grd <- t(param.grd)
  if(length(area) != 1){
    simul <- TUWmodel(prec=prec, airt=tmean, ep=pet,
                      area=area, param=param.grd)
    simul <- zoo(simul$q,date)
  }
  if(length(area) == 1){
    simul <- TUWmodel(prec=as.numeric(prec), airt=as.numeric(tmean),
ep=as.numeric(pet),
                      area=area, param=param.grd)
```

```
   simul <- zoo(t(simul$q),date)
  }
  simul[is.na(simul)] <- -999
  SIMUL_PASS[[NOMI[i]]] <- simul

# ---------------------- PLOT MAPS FOLLOWING FLOODING EVENTS --------------------- #

Piemonte <- st_union(qualibacini$geometry)
CUT_RIVERS <- vector('list',length(Final_CAT))
ID <- NOMI
names(CUT_RIVERS) <- ID

#Elimination rivers' section exceeding regional borders
for(i in 1:length(CUT_RIVERS)){
  cat(i)
  timestamp()
  CUT_RIVERS[[i]] <- st_intersection(Piemonte,Final_CAT[[i]]$FIUMI)
}

#Convert simulated discharges as matrix
Q <- matrix(NA,ncol=length(date),nrow=length(SIMUL_PASS))
for(i in 1:length(SIMUL_PASS)){
  Q[i,] <- SIMUL_PASS[[i]]
}

ID <- names(SIMUL_PASS)
Event <- rep(NA,6)                                    #Matrix for the flood event
s <- "2020-09-30"                                     #Start date
e <- "2020-10-05"                                     #End date
yy <- 2020                                            #Year
start <- which(date == s)
end <- which(date == e)

#Estraction discharges for the event
for(i in 1:length(SIMUL_PASS)){
  cat(i)
  timestamp()
  dummy <- as.numeric(SIMUL_PASS[[i]][start:end])
  Event <- rbind(Event,dummy)
}
Event <- Event[-1,]
colnames(Event) <- as.character(seq(from=as.Date(s),to=as.Date(e),by="day"))
rownames(Event) <- ID

pal <- colorRampPalette(c("lightblue","red"))     #Definition of the colorscale
min <- 0                                          #Minimum discharge for colorscale
max <- 100                                        #Maximum discharge for colorscale
ncol <- 100                                       #Number of intervals of the colorscale
int <- (max-min)/ncol                             #Intervals width definition
col <- pal(ncol)                                  #Colors definition
cuts <- seq(from = min,to = max, by = int    #Intervals edges definition


# ---------------------------------- MAPS PLOTS ---------------------------------- #
setwd("/saving_destination")              #Set the path where to save the .pdf
par(mai=c(0.1,0.1,0.2,0.1))               #Graphical settings

for(j in 1:dim(Event)[2]){
  n <- paste("Flood_",yy,"_",j,".pdf",sep="")           #Title of .pdf
  pdf(file=n,width = 9, height = 9)                     #.pdf creation
```

```
  #Start plot
  plot(Piemonte,main = colnames(Event)[j],cex.main = 0.8,border="azure4")         #
  for (i in 1:length(ID)){
    flag1 <- FALSE
    for(k in 1:(length(cuts)-1)){
      if(flag1 == FALSE){
        if(between(Event[i,j],cuts[k],cuts[k+1]) == TRUE){

          flag1 <- TRUE
          flag2 <- k
        }
      }
    }

    flag3 <- FALSE
    for(k in 1:(length(wd_a)-1)){
      if(flag3 == FALSE){
        if(between(AREA[i],wd_a[k],wd_a[k+1]) == TRUE){
          flag3 <- TRUE
          flag4 <- wd[k]
          TRNSP <- tsnp[k]
        }
      }
    }
    plot(CUT_RIVERS[[i]],col=col[flag2],lwd=flag4,add=TRUE)
  }
  dev.off()                                                          #End plot
}


# ------------------ SEPARATE CREATION OF THE COLORSCALE GRAPHIC --------------- #
layout(matrix(c(1,1,2,2),byrow=TRUE))
plot(1)
legend_image <- as.raster(matrix(pal(100), nrow=1))
plot(c(0,2),c(0,1),type = 'n', axes = F,xlab = '', ylab = '', main = 'mm/d')
text(x=1.5, y = seq(0,100,l=20), labels = seq(0,100,l=20))
rasterImage(legend_image, 0, 0, 2,2)
```

BIBLIOGRAPHY/SITOGRAPHY

Addor, N., Nearing, G., Prieto, C., Newman, A. J., Le Vine, N., & Clark, M. P., (2018), *A ranking of hydrological signatures based on their predictability in space*, Water Resources Research, 54, 8792–8812. https://doi.org/10.1029/2018WR022606.

Akbari, A., Ng, L., & Solnik, B., (2021), *Drivers of economic and financial integration: A machine learning approach*, Journal of Empirical Finance, 61. 10.1016/j.jempfin.2020.12.005.

ARPA Piemonte, (2008), *Rapporto Preliminare sull'Evento Alluvionale del 28-30 Maggio 2008*, http://www.regione.piemonte.it/cgi-bin/montagna/pubblicazioni/frontoffice/elenco.cgi?id_settore=10&flag_cambio_area=&tipo=&area=10&argomento=111.

Beck, M. B., (2002), *Environmental foresight and models: A manifesto*, Oxford: Elsevier

Beven, K., (1975), *A deterministic, spatially distributed model of catchment hydrology*, University of East Anglia.

Beven, K., Warren, R., & Zaoui, J., (1980), *SHE: Towards a methodology for physically-based distributed forecasting in hydrology*, Hydrological Forecasting, The Oxford Symposium, IAHS-AISH, 133–137.

Beven, K., (1993), *Prophecy, reality and uncertainty in distributed hydrological modelling*, Advances in Water Resources, 16(1), 41–51. https://doi.org/10.1016/0309-1708(93)90028-E.

Beven, K., (2006), *A manifesto for the equifinality thesis*, Journal of Hydrology, 320(1–2), 18–36. https://doi.org/10.1016/j. jhydrol.2005.07.007.

Blöschl, G., and Sivapalan, M., (1995), *Scale issues in hydrological modelling – A review*., Hydrol. Process., 9(3–4), 251–290.

Bloschl, G., Sivapalan, M., Wagener, T., Viglione, A., & Savenije, H. (Eds.). (2013). *Runoff Prediction in Ungauged Basins: Synthesis across Processes, Places and Scales.* Cambridge: Cambridge University Press. Doi:10.1017/CBO9781139235761.

Boi M., Orrù A., (2019), *Carta del Curve Number Regionale*, Agenzia Regionale per la Protezione dell'Ambiente della Sardegna - ARPAS

Brouwer C., Heibloem M., (1986), *IRRIGATION WATER MANAGEMENT Training manual no. 3,* Part II - Chapter 3, FAO.

Budyko, M. I., (1974), *Climate and life*, International Geophysics Series, 18: 508.

Chow, T. V., Maidment, R. D., & Mays, W. L., (1988), *Applied hydrology (2010 ed.)*, New York, NY: Tata McGraw-Hill.

Dwarakish, G.S. & Bigganahalli Puttaswamigowda, Ganasri, (2015), *Impact of land use change on hydrological systems: A review of current modeling approaches*, Cogent Geoscience. 1. 1115691. 10.1080/23312041.2015.1115691.

European Environment Agency (EEA), (2012), *Technical report No 7/2012 - Rationales, building and improving for widening uses to Water Accounts and WISE applications*, https://www.eea.europa.eu/publications/eea-catchments-and-rivers-network.

Feyen, L., Vázquez, R., Christiaens, K., Sels, O., & Feyen, J., (2000), *Application of a distributed physically-based hydrological model to a medium size catchment*, Hydrology and Earth System Sciences, 4, 47–63. http://dx.doi.org/10.5194/hess-4-47-2000.

Food And Agriculture Organization of the United Nations (FAO), (2006), *Guidelines for soil description, fourth edition.*

Gallo E., Ganora D., Laio F., Masoero A., & Claps P., (2013), *Atlante dei bacini imbriferi piemontesi (Atlas of river basins in Piemonte)*.

Gupta, H. V., Kling, H., Yilmaz, K. K., and Martinez, G. F., (2009), *Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling*, Journal of Hydrology., 377, 80–91, https://doi.org/10.1016/j.jhydrol.2009.08.003.

Khatami, S., Peel, M. C., Peterson, T. J., & Western, A. W., (2019), *Equifinality and flux mapping: A new approach to model evaluation and process representation under uncertainty*, Water Resources Research, 55, 8922–8941. https://doi.org/10.1029/ 2018WR023750.

Knoben, W. J. M., Freer, J. E., and Woods, R. A., (2019), *Technical note: Inherent benchmark or not? Comparing Nash–Sutcliffe and Kling–Gupta efficiency scores*, Hydrology and Earth System Sciences, 23, 4323–4331, https://doi.org/10.5194/hess-23-4323-2019.

Merz, R., Tarasova, L., & Basso, S. (2020), *Parameter's controls of distributed catchment models—How much information is in conventional catchment descriptors?*, Water Resources Research, 56, e2019WR026008. https://doi.org/10.1029/2019WR026008.

Mills P., Nicholson O., Reed D., (2014), *Physical Catchment Descriptors – Volume IV*.

Mishra, A. K., and Coulibaly, P. (2009), *Developments in hydrometric network design: A review*, Rev. Geophys., 47, RG2001, doi:10.1029/2007RG000243.

Mullen, K. M., Ardia, D., Gil, D. L., Windover, D., & Cline, J., (2011), *DEoptim: An R Package for Global Optimization by Differential Evolution*, Journal of Statistical Software, *40*(6), 1–26. https://doi.org/10.18637/jss.v040.i06

Nachtergaele F., van Velthuizen H., Verelst L., (2009), *Harmonized World Soil Database*, FAO.

Nandakumar, N., & Mein, R. G., (1997), *Uncertainty in rainfall– runoff model simulations and the implications for predicting the hydrologic effects of land-use change*, Journal of Hydrology, 192, 211–232, http://dx.doi.org/10.1016/S0022-1694(96)03106-X.

Nash, J. E. and Sutcliffe, J. V., (1970), *River flow forecasting through conceptual models part I – A discussion of principles*, Journal of Hydrology, 10, 282–290, https://doi.org/10.1016/0022-1694(70)90255-6.

Niel, H., Paturel, J. E., & Servat, E., (2003), *Study of parameter stability of a lumped hydrologic model in a context of climatic variability*, Journal of Hydrology, 278, 213–230, http://dx.doi.org/ 10.1016/S0022-1694(03)00158-6.

Parajka, J., Merz, R., & Blöschl, G., (2007), *Uncertainty and multiple objective calibration in regional water balance modelling: Case study in 320 Austrian catchments*, Hydrological Processes. 21. 435 - 446. 10.1002/hyp.6253.

R Core Team, (2022), *Writing R Extensions, Version 4.2.1 (2022-06-23).*

Pesce, M., Viglione, A., von Hardenberg, J., Tarasova, L., Basso, S., and Merz, R., (2022), *HydroPASS: a newly developed R package to go through the regional calibration of distributed catchment models*, EGU General Assembly 2022, Vienna, Austria, 23–27 May 2022, EGU22-569, https://doi.org/ 10.5194/ egusphere-egu22-569.

Razavi T., Coulibaly P., (2013), *"Streamflow Prediction in Ungauged Basins: Review of Regionalization Methods"*, Journal of Hydrologic Engineering, 18, 958-975, 10.1061/ (ASHE)HE. 1943-5584.0000690.

Regione Piemonte, (1998), *Eventi Alluvionali in Piemonte, 2-6 Novembre 1994, 8 Luglio 1996, 7-10 Ottobre 1996*, http://www.regione.piemonte.it/cgi-bin/montagna/pubblicazioni/ frontoffice/ elenco.cgi?id_settore=10&flag_cambio_area=&tipo=&area=10&argomento=111.

Regione Piemonte, (2001), *Evento alluvionale del 13-16 ottobre 2000 in Piemonte - Analisi meteorologica e idrologica*, http://www.regione.piemonte.it/cgi-bin/montagna/pubblicazioni/ frontoffice/elenco.cgi?id_settore=10&flag_cambio_area=&tipo=&area=10&argomento=111.

Regione Piemonte, *Eventi alluvionali in Piemonte - ANALISI METEOROLOGICA E IDROLOGICA*, http://www.regione.piemonte.it/cgi-bin/montagna/pubblicazioni/frontoffice/elenco.cgi?id_settore= 10&flag_cambio_area=&tipo=&area=10&argomento=111.

Regione Piemonte, (2020*), Evento Alluvionale del 2-3 Ottobre 2020*, https://www.Regione.piemonte. it/ web/pubblicazioni-editoriali?titolo=&destinatari=All&tema=255&sort_by=field_contatore_ visualizzazioni_value.

Robinson, N., Regetz, J., and Guralnick, R. P., (2014), *EarthEnv-DEM90: A nearly-global, void-free, multi-scale smoothed, 90m digital elevation model from fused ASTER and SRTM data*, ISPRS Journal of Photogrammetry and Remote Sensing, 87:2014, 57-67, http://www.sciencedirect.com/science /article/pii/S0924271613002360. Download of the DTM at https://www.earthenv.org/DEM.

Rogelis, M. C., Werner, M., Obregón, N., and Wright, N., (2016), *Hydrological model assessment for flood early warning in a tropical high mountain basin*, Hydrology and Earth System Sciences, https://doi.org/10.5194/hess-2016-30.

Schreider, S. Y., Jakemanb, A. J., Gallant, J., and Merritt, W. S. (2002), *Prediction of monthly discharge in ungauged catchments under agricultural land use in the Upper Ping basin, northern Thailand*, Math. Comp. Sim., 59(1–3), 19–33.

Sivapalan M., Takeuchi K., Franks S., Gupta V. K., Karambiri H., Lakshmi V., Liang X., McDonnel J. J., Mendiondo E. M., O'Connel P. E., Oki T., Pomeroy J. W., Schertzer D., Uhlenbrook S. & Zehe E. (2003*), "IAHS Decade on Predictions in Ungauged Basins (PUB), 2003–2012: Shaping an exciting future for the hydrological sciences"*, Hydrological Sciences Journal, 48:6, 857-880, DOI: 10.1623/ hysj.48.6.857.51421.

Venables W. N., Smith D. M., and R Core Team, (2022), *Notes on R: A Programming Environment for Data Analysis and Graphics Version 4.2.1 (2022-06-23).*

Wagener, T., and Wheater, H. S., (2006), *Parameter estimation and regionalization for continuous rainfall-runoff models including uncertainty*, Journal of Hydrology, 320(1–2), 132-154

Young A. R., (2006), *"Stream flow simulation within UK ungauged catchments using a daily rainfall-runoff model",* Journal of Hydrology, Volume 320, Issues 1–2, Pages 155-172, ISSN 0022-1694, https://doi.org/10.1016/j.jhydrol.2005.07.017.

ISO 14688-1:2002, (2002), *Geotechnical investigation and testing — Identification and classification of soil — Part 1: Identification and description*.

[Dataset] ARPA Piemonte*, Optimal Interpolation Database, http://www.idrologia.polito.it /web2 /open-data/ cd_Dati_Regione_ Piemonte/50_anni_dati_meteo_Piemonte/index.htm.*

[Dataset] ARPA Piemonte (dipartimento sistemi previsionali), (2010), *Metodologia, http:// www. idrologia. polito.it/web2/open-data/cd_Dati_Regione_Piemonte/50_anni_dati_meteo_Piemonte /metodologia.htm.*

[Dataset] CORINE Land Cover, Copernicus – Land Monitoring Service, https://land. copernicus.eu/ pan-european/corine-land-cover.

[Dataset] European catchments and Rivers network system (Ecrins), https://www.eea.europa.eu/data-and-maps/data/european-catchments-and-rivers-network#tab-gis-data.

[Dataset] Harmonized World Soil Database (HWSD) v 1.2, FAO, https://www.fao.org/soils-portal/ data-hub/soil-maps-and-databases/harmonized-world-soil-database-v12/en/.

[Dataset] Piedmont "Ambiti Ammministrativi" (for regional boundary shape file determination).