## 1.2   Interpolation of meteorological input data

WaSiM-ETH provides various methods to interpolate input data with a sparse spatial resolution to the model grid resolution. Usually, input data are given as time series of climate stations or as sparse input grids or pseudo station time series of climate models. The interpolation can be carried out by using one or more of the methods described below. As new feature starting with WaSiM-ETH 8.05.00 multiple interpolation methods may be linearly combined with different weights for different regions, thus offering a great flexibility.

The available methods are (numbers refer to the method-ID used in the control file):
1.  Inverse Distance Interpolation (IDW). Use: for precipitation, sunshine duration and other data which are not (strongly) dependent on elevation or when modelling flat regions without substantial elevation ranges
2.  Elevation Dependent Regression with External Preprocessing (EDREXT). Use: for all elevation dependent types of input data like temperatures, vapor pressure, air humidity, wind speed. Recommended for basins with a substantial elevation range only
3.  A linear combination of IDW + EDREXT. Use: for data that depend only partly on elevation, e.g. precipitation in high mountain areas.
4.  Thiessen Polygones. Use: if only one station is available (same results as IDW or EDR but faster)
5.  Bilinear Interpolation (BLI). Use: when using gridded time series from RCM or GCM output (WaSiM-tables containing time series for each grid cell of the GCM run)
6.  Bilinear Interpolation of both gradients and residuals and a linear combination thereof (BIGRES). Use: same as BLI but for data which are separately available as residues and gradients
7.  Bicubic Spline Interpolation (BSI). Use: same as BLI but with another (smoother) technique
8.  Bicubic Splice Interpolation of both gradients and residuals and a linear combination thereof (BSIRES). Use: same as BIGRES, but somewhat smoother
9.  A pseudo-method: Reading externally interpolated data from grids; Use: when gridded input is already available (in the model raster dimensions!) from external interpolation routines
10. Elevation dependent regression with internal processing (EDRINT); like EDR but without external preprocessing using regr or regress. Use: same as method 2. The main difference between methods 2 and 10 is, that for method 2 the stations to be used can be selected manually in the ini file for the regress tool wheres for method 10 all stations in the input file will be used. Also, the internal preprocessing does not allow the mixing of input files with different temporally resolution as it is possible in regr and regress. To achieve identical results, the input file used for method 10 should only contain the stations which were used in the regress-ini-file, too.
11. A linear combination of IDW and EDRINT, like IDW+EDREXT but without external preprocessing using regr or regress. Use: same as for method 3; read comments on method 10 hereabove.

Each of these methods is described in detail in the following chapters.

### 1.2.1   Inverse Distance Interpolation (Method 1)

Method ID for all methods using this algorithm: 1, 3, and 11

For distance weighted interpolation there are two possibilities offered in WaSiM-ETH: the inverse distance weighting method and the Thiessen-polygone method, whereby Thiessen-polygones can be seen as a special case of IDW using only the nearest station for getting an "interpolation" result.

The input data are read in as a WaSiM-table. Regression gradient files cannot be processed since they contain no station data but only gradients. For the IDW-method all stations within a specified search radius are used for the interpolation. If all stations have identical values, the interpolation is skipped and the constant value is taken instead. If all stations have nodata values, the interpolation is skipped and the results from the last time step is taken instead.

The interpolation result is the sum of all contributing weighted station data:

$$\hat{z}(u) = \sum_j \left( w_j \cdot z(u_j) \right) \tag{1.2.1}$$

$$\text{with} \quad w_j = \frac{1}{d(u,u_j)^p} \cdot \frac{1}{C} \quad \text{and} \quad C = \sum_j \frac{1}{d(u,u_j)^p} \quad \text{follows:} \quad \sum_j w_j = 1.0$$

with $\hat{z}(u)$    interpolated value at location u
       $w_j$         weight of the observed value at the station j
       $z(u_j)$      observed value at the station j
       $d(u,u_j)$   distance to the station j
       $p$          weighting power of the inverse distance (between 1 and 3, 2 is recommended)

WaSiM-ETH offers the possibility to change the parameter p, to specify a maximum distance dmax as well as two parameters specifying the anisotropy, which can be seen as an ellipse with a specified slope of the main axis against the horizontal (west-east-line, using mathematically positive rotation orientation) and with a specified length ratio of the long (main) axis to the short axis. The slope of the main axis may range from 0 to <90°, the axis length ratio may range from greater than zero to 1 (the latter would indicate that there is no anisotropy at all).

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
1                            # method
$inpath//regen//$year//.dat AdditionalColumns=0    # station data
$inpath//regen//$year//.out  # regression data (if method = 2 or 3)
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid                   # writecode
0.1                          # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998                          # error value
$IDWweight        # weighting of the reciprocal distance for IDW
0.75             # for method 3: relative weight of IDW
$IDWmaxdist       # max. distance of stations
$Anisoslope       #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie      #  ratio short to long axis of anisotropy-ellipsis
0.1                  # lower limit of interpolation results
0                    # replace value for results below the lower limit
900                  # upper limit for interpolation results
900                  # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3        # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1        # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

**Examples:**

Figures 1.2.1 and 1.2.2 show typical examples of pure IDW and pure EDR interpolations for the

same date (January 3, 1994, 17:00). To get the most information out of the data, a combination of IDW and EDR can be used → see chapter 1.2.3.
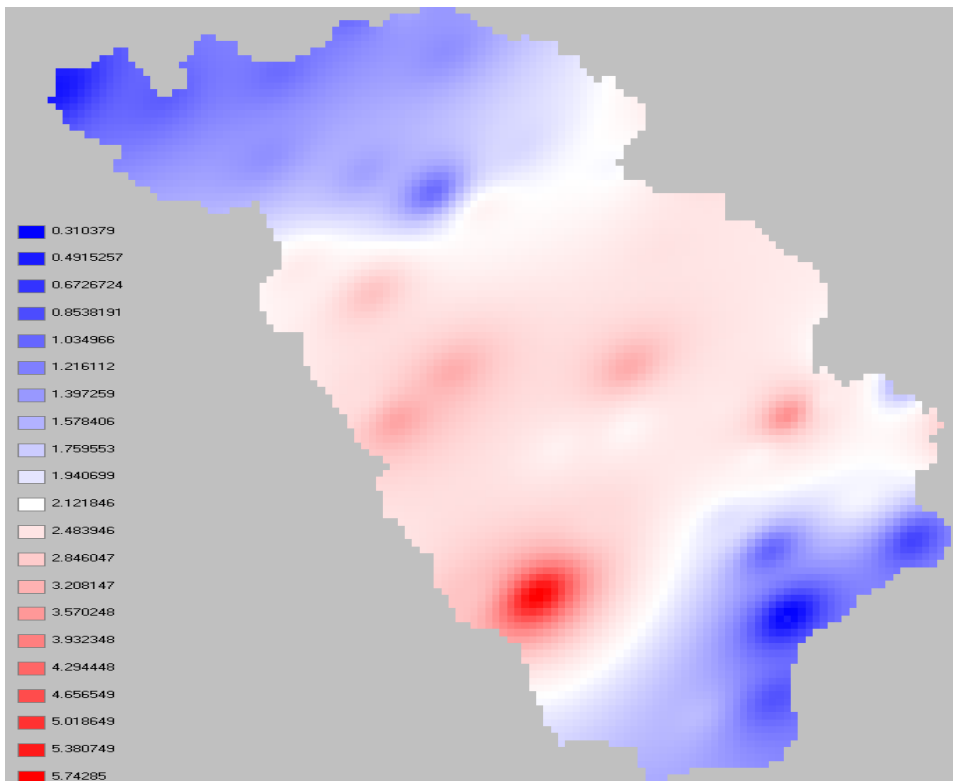


| | |
|---|---|
| ■ | 0.310379 |
| ■ | 0.4915257 |
| ■ | 0.6726724 |
| ■ | 0.8538191 |
| ■ | 1.034966 |
| ■ | 1.216112 |
| ■ | 1.397259 |
| ■ | 1.578406 |
| ■ | 1.759553 |
| ■ | 1.940699 |
| ■ | 2.121846 |
| ■ | 2.483946 |
| ■ | 2.846047 |
| ■ | 3.208147 |
| ■ | 3.570248 |
| ■ | 3.932348 |
| ■ | 4.294448 |
| ■ | 4.656549 |
| ■ | 5.018649 |
| ■ | 5.380749 |
| ■ | 5.74285 |

*Figure 1.2.1: example for IDW interpolation of precipitation (January 3, 1994, 18:00)*



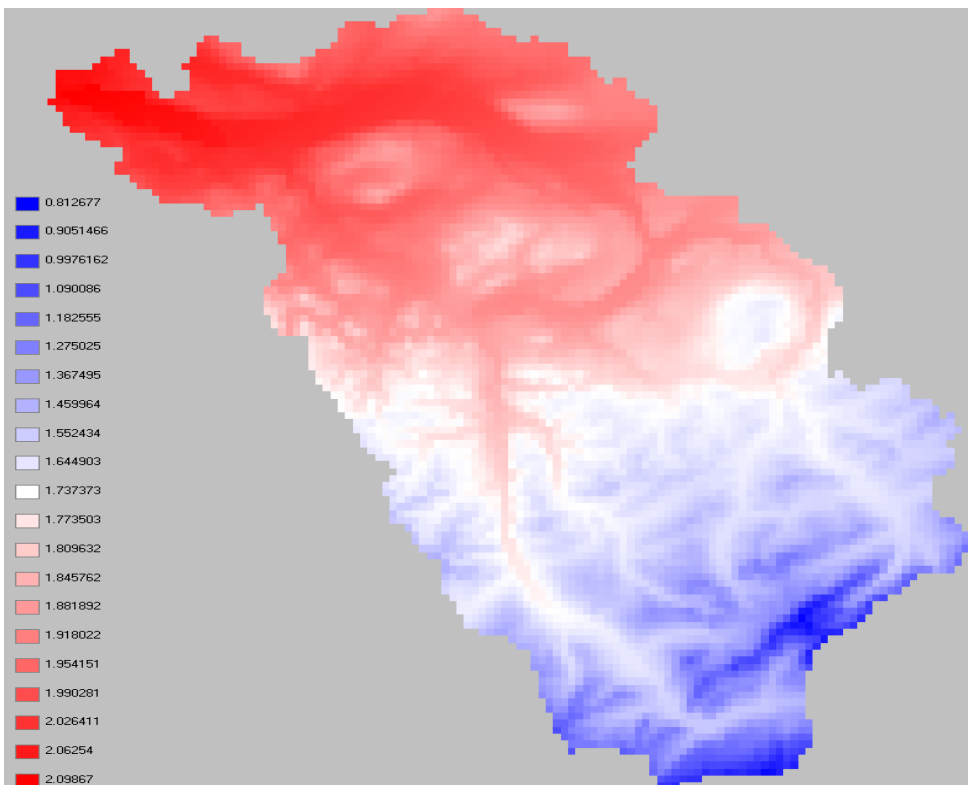| | |
|---|---|
| ■ | 0.812677 |
| ■ | 0.9051466 |
| ■ | 0.9976162 |
| ■ | 1.090086 |
| ■ | 1.182555 |
| ■ | 1.275025 |
| ■ | 1.367495 |
| ■ | 1.459964 |
| ■ | 1.552434 |
| ■ | 1.644903 |
| ■ | 1.737373 |
| ■ | 1.773503 |
| ■ | 1.809632 |
| ■ | 1.845762 |
| ■ | 1.881892 |
| ■ | 1.918022 |
| ■ | 1.954151 |
| ■ | 1.990281 |
| ■ | 2.026411 |
| ■ | 2.06254 |
| ■ | 2.09867 |

*Figure 1.2.2: example for EDR interpolation for the same data as in figure 1.2.1*

### 1.2.2 Elevation dependent regression (Method 2)

Method ID for all methods using this algorithm: 2, 3, 10 and 11

Using this method, the interpolation result is estimated using the elevation dependence of a variable. If there are data of more than three station available, an additional trend surface is estimated in order to correct shifts in the horizontal directions. The trend surface uses only the residuals of the elevation dependent regression. To use this method within WaSiM it is either necessary to generate the parameter files containing the elevation gradients for each of these variables using the programs "regress" or "regr" or to use the internal regression methods.

The gradients will be estimated for at maximum three elevation ranges separated by inversions from each other. The estimation of the elevation of these inversions is done semi automatically by searching for intersections of the regressions of neighboring elevation ranges. As initial values typical elevation ranges of inversions has to be specified for each of the two possible inversions. Separate regressions are then estimated for the ranges below and above the lower inversion. If there is an intersection of both regressions between the lower and the upper (theoretical) inversion (plus an adjustable tolerance), then it is assumed that there is only one inversion at all. Otherwise, i.e. if the regressions are crossing outside of the range between the two inversions (plus tolerance), a twofold inversion is assumed making the lower regression valid below the lower inversion only and the upper regression above the upper inversion. For the elevation range between both inversions a third gradient is assumed to link both regressions linearly.

The interpolation to a specified spot in the model is then done by :

$$T(h_M) = a_i + b_i \cdot h_M \tag{1.2.2}$$

with  $h_M$       elevation [m]
         $T$        variable (e.g. air temperature)
         $i$         index for lower, medium resp. upper regression (i=1..3)
         $a_i, b_i$    parameter of the respective regression (i=1...3)

For the actual sample, parameters $a$ and $b$ are estimated as $\hat{a}$ and $\hat{b}$ using a least square fit (linear regression):

$$g(\hat{a}, \hat{b}) = \sum_i \left(T_i - \hat{a} - \hat{b} \cdot h_{M,i}\right)^2 \to Min \tag{1.2.3}$$

The partial derivatives of (1.2.3) for $\hat{a}$ and $\hat{b}$ result in a linear equation system with $\hat{a}$ and $\hat{b}$ as the unknowns, which may be estimated by:

$$\hat{a} = \frac{1}{n} \cdot \left(\sum T_i - \hat{b} \sum h_{M,i}\right) \quad \text{and} \quad \hat{b} = \frac{n \cdot \sum T_i h_{M,i} - \sum T_i \sum h_{M,i}}{n \cdot \sum h_{M,i}^2 - \left(\sum h_{M,i}\right)^2} \tag{1.2.4}$$

By applying equation (1.2.4) to all elevation data of the actual sample and calculating the difference between this regression value the actual value of each sample data, the sample's residues $r_i$ with respect to the regression range are calculated:

$$r_i = \begin{cases} T_i - (a_3 + b_3 \cdot h_{M,i}) + a_1 & h_{M,i} > h_{v,u} \\ T_i - (a_2 + b_2 \cdot h_{M,i}) + a_1 \quad \text{for} & h_{v,l} < h_{M,i} \leq h_{v,u} \\ T_i - (b_1 \cdot h_{M,i}) & h_{M,i} \leq h_{v,l} \end{cases} \tag{1.2.5}$$

with   $r_i$      residuum
       $T_i$      observation value I (i=1...n; n=number of values)

$a_{1...3}$       regression parameters $\hat{a}$ after (1.2.3) for the elevation ranges 1 to 3
$b_{1...3}$       regression parameters $\hat{b}$ after (1.2.3) for the elevation ranges 1 to 3
$h_{M,i}$       elevations of stations used for sample data
$h_{v,l}$       lower inversion (given as a parameter in m a.s.l.)
$h_{v,u}$       upper inversion (given as a parameter in m a.s.l.)

**Note**: For perfomance optimization, the value of $a_1$ is added to all $r_i$-values. Thus, the application of the regression parameters for the later interpolation is somewhat faster because most grid cells will be below the lower inversion usually (and by eleminating the offset $a_1$, two additions are saved)

To fit a trend surface through these residues, a multiple linear regression is carried out:

$$r = u + v_1 x + v_2 y \tag{1.2.6}$$

with     $r$       fitted residuum for location $x,y$
        $u,v_1,v_2$ regression parameters for the multiple linear regression
        $x,y$     coordinates of the station location with respect to the grid coordinates

By minimizing the square error, the parameters $u$, $v_1$ and $v_2$ are estimated as $\hat{u}$, $\hat{v}_1$ and $\hat{v}_2$ :

$$\hat{v}_1 = \frac{1}{D} \cdot \begin{vmatrix} n & \sum r & \sum y \\ \sum x & \sum rx & \sum xy \\ \sum y & \sum ry & \sum y^2 \end{vmatrix} \qquad \hat{v}_2 = \frac{1}{D} \cdot \begin{vmatrix} n & \sum x & \sum r \\ \sum x & \sum x^2 & \sum rx \\ \sum y & \sum xy & \sum ry \end{vmatrix} \tag{1.2.7}$$

$$\text{with} \qquad D = \begin{vmatrix} n & \sum x & \sum y \\ \sum x & \sum x^2 & \sum xy \\ \sum y & \sum xy & \sum y^2 \end{vmatrix} \qquad \text{and} \qquad \hat{u} = \bar{r} - \hat{v}_1 \bar{x} - \hat{v}_2 \bar{y} \tag{1.2.8}$$

By using the parameter $\hat{a}_{1...3}, \hat{b}_{1...3}, \hat{u}, \hat{v}_1$ and $\hat{v}_2$ the interpolation can be carried for each elevation $h_{M,i}$ and for each set of coordinates $x$ and $y$:

$$T_r = \begin{cases} \hat{u} + \hat{v}_1 x + \hat{v}_2 y - \hat{a}_1 + \hat{a}_3 + \hat{b}_3 z & z > h_{v,u} \\ \hat{u} + \hat{v}_1 x + \hat{v}_2 y - \hat{a}_1 + \hat{a}_2 + \hat{b}_2 z & \text{for} \quad h_{v,l} \leq z \leq h_{v,u} \\ \hat{u} + \hat{v}_1 x + \hat{v}_2 y + \hat{b}_1 z & z < h_{v,l} \end{cases} \tag{1.2.9}$$

with     $T_r$       interpolated value
        $z$       elevation at coordinate $x,y$

**Note**: since all residues where extended by the value of $\hat{a}_l$, the multiple linear regression in equation (1.2.9) has to account for this offset by subtracting $\hat{a}_1$ from all regressions, resulting in a faster calculation for all grid cells below the lower inversion boundary $h_{v,l}$ (last row in (1.2.9)).

### Practical application

### Control file:

the following example shows the settings for an interpolator using elevation dependent regression. There are two variants shows: first the traditional method using external preprocessing and second the new version with internal regression processing. The important parameters for elevation dependent regression are written in a red, bold, enlarged font. See also description of the control file in this manual for more information about the other parameters:

```
[temperature]
2  # method, here 2 for regression with external preprocessing
$inpath//tempe//$year//a.dat AdditionalColumns=0 # station data
$inpath//temper//$year//.out # file name with regression data
$outpath//reg_//$tempegrid   # name of the output grid
$Writegrid                   #
0.1                          # correction faktor for results
$outpath//reg_temp//$grid//.//$code//$year $hour_mean # statistics file
998                          # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.1             # relative weight of IDW-interpolation
$IDWmaxdist     # max. distance of stations
$Anisoslope     # slope of mean axis of the anisotropy-ellipsis
$Anisotropie    #  ratio short to long axis of the anisotropy-ellipsis
-40                  #  lower limit of interpolation results
-40                  #  replace value for results below the lower limit
40                   #  upper limit for interpolation results
40          #  replace value for results with larger values than the upper limit
$SzenUse    #  1=use scenario data for correction, 0=dont use scenarios
1           # 1=add scenarios, 2=multiply scenarios, 3=percentual change
4           #  number of scenario cells
...
```

and here the example for method 10 (and 11) with internal regression (pre-)processing

```
[temperature]
10   # method, here 10 for regression with internal preprocessing
$inpath//tempe//$year//a.dat AdditionalColumns=0 # station data
820 1400 200 1 300 # lower inversion [m asl], upper inversion
# [m asl], tolerance [m], overlap [0/1 for true/false],
# clusterlimit [m]
$outpath//reg_//$tempegrid   # name of the output grid
$Writegrid                   #
0.1                          # correction faktor for results
$outpath//reg_temp//$grid//.//$code//$year $hour_mean # statistics file
998                          # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.1             # relative weight of IDW-interpolation
$IDWmaxdist     # max. distance of stations
$Anisoslope     # slope of mean axis of the anisotropy-ellipsis
$Anisotropie    #  ratio short to long axis of the anisotropy-ellipsis
-40                  #  lower limit of interpolation results
-40                  #  replace value for results below the lower limit
40                   #  upper limit for interpolation results
40          #  replace value for results with larger values than the upper limit
$SzenUse    #  1=use scenario data for correction, 0=dont use scenarios
1           # 1=add scenarios, 2=multiply scenarios, 3=percentual change
4           #  number of scenario cells
```

When using methods 10 or 11, there will be no name for an externally produced parameter file with regression coefficients. Instead, the most important parameters must be given there. Their meaning is as follows (compare description of regression programs "regr" and "regress"):

- 1st parameter: lower inversion in m (above sea level)

- 2nd parameter: upper inversion in m (above sea level)

- 3rd parameter: tolerance (when the upper and the lower regression intersect within $[h_{v,l}$-tolerance … $h_{v,u}$+tolerance], then only two regressions are used, otherwise three regressions will be used.

- 4th parameter: boolean value, 0 for strictly separating both elevation ranges when estimating the regression parameters, 1 for allowing an overlap: the lowest station of the upper elevation range (above lower inversion) will be used for the lower regression and the highest station of the lower regression range (below lower inversion) will be used for the upper regression, thus reducing the risk of to sharp changes in gradients when the number of stations is relatively low in one or both elevation ranges

- 5th parameter: minimum elevation spread for the stations in one elevation range to be really used for a linear regression – if all stations fall within this range (i.e. the difference in elevations of the highest station and the lowest station in one elevation range is less that this parameter), then all stations will be handled as a cluster (all station data of this range will be averaged → this means, that usually no real regression will be done but constant values for the entire elevation range will be used.

**Examples:**

When using the tool regress, the elevations of the inversions can be estimated interactively (by trial and error). A typical screen output looks like the one shown in figure 1.2.2.
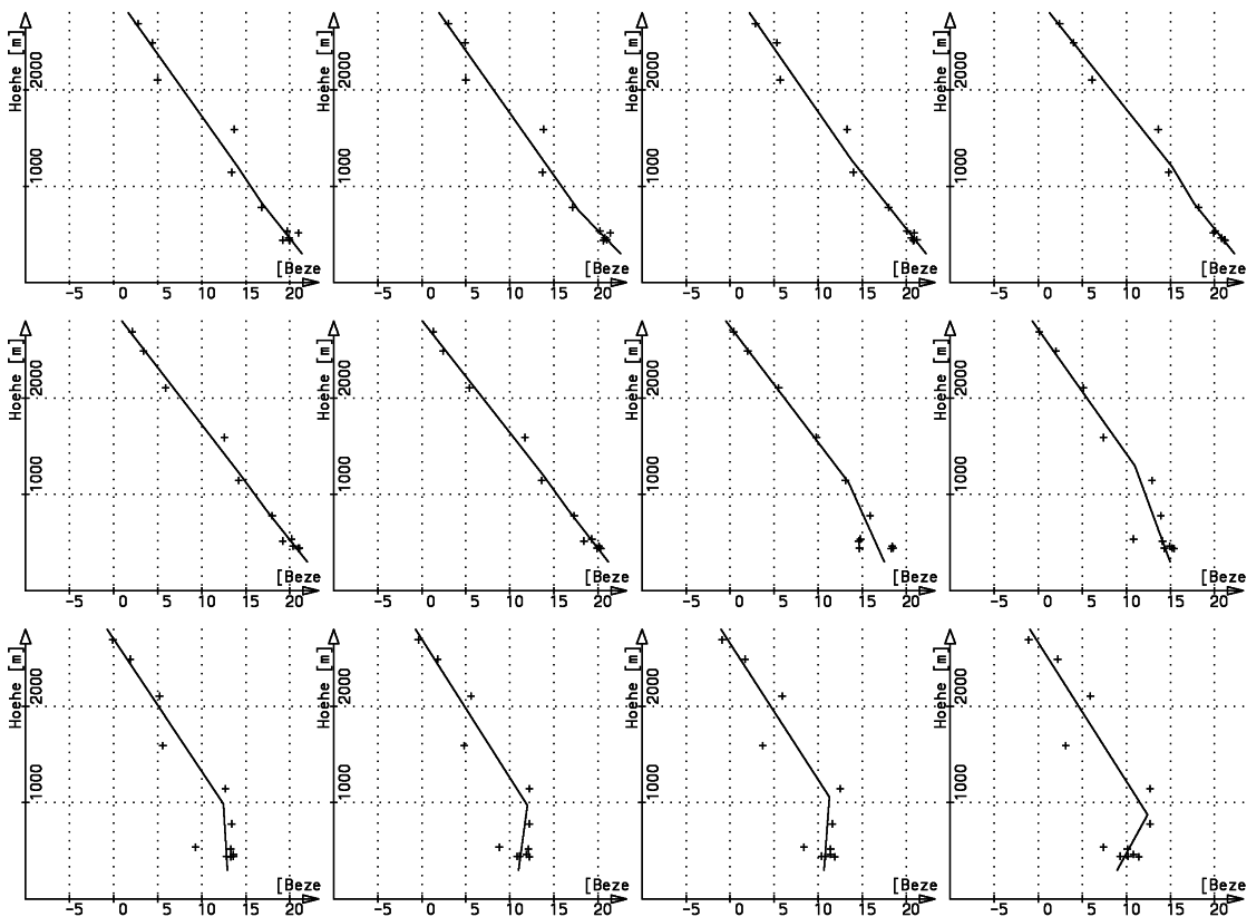


*Figure 1.2.3: typical temperature gradients (here for May 5, 1996, hours 1 to 12)*

In Figure 1.2.1, there are shown both cases: The upper right hand graph shows the temperature with three regression ranges because the uppermost and the lowermost regressions will not intersect between 820m and 1200m (although only barely visible here) whereas the lower right hand graph shows an example with only two active elevation ranges (because both regressions intersect between the upper and lower inversion.

The resulting lines in the preprocessing output file look like this (formatted with an additional header for convenience):

for the upper left hand example (measures in 1/10°C or m or (1/10°C)/m):

```
yy    mm dd hh   a1        b1       hv,l    a2        b2       hv,u     a3
1996  5  6  16   248.462  -0.08697  820.0   232.154  -0.06709  1200.0   255.908
 b3       u         v1                         v2
 -0.08688 145.71315 1.33328886195307E-0004 2.77000081244013E-0005
```

To calculate the temperature for a given cell with an elevation of 1000m and coordinates of x=731000 and y=248000, the second regression must be used (as well as parameter $a_1$ as described in the notes to equations 1.2.5 and 1.2.9). Using the parameters here, results in:

$$T = \hat{u} + \hat{v}_1 x + \hat{v}_2 y - \hat{a}_1 + \hat{a}_2 + \hat{b}_2 z$$
$$= 145.713 + 0.0001333 \cdot 731000 + 0.0000277 \cdot 248000 - 248.462 + 232.154 - 0.06709 \cdot 1000$$
$$= 166.62 = 16.62°C$$

Figure 1.2.2 shows an example of elevation dependent regression for the first day of January 1998, hour 14. The high mountains in the South East show the deepest temperatures, the hilly regions have relatively high values and the "low lands" (~400...500m) in the North section have low values again: this is the typical pattern of a temperature inversion as shown in figure 1.2.1 in the lower left hand chart (but for another date).
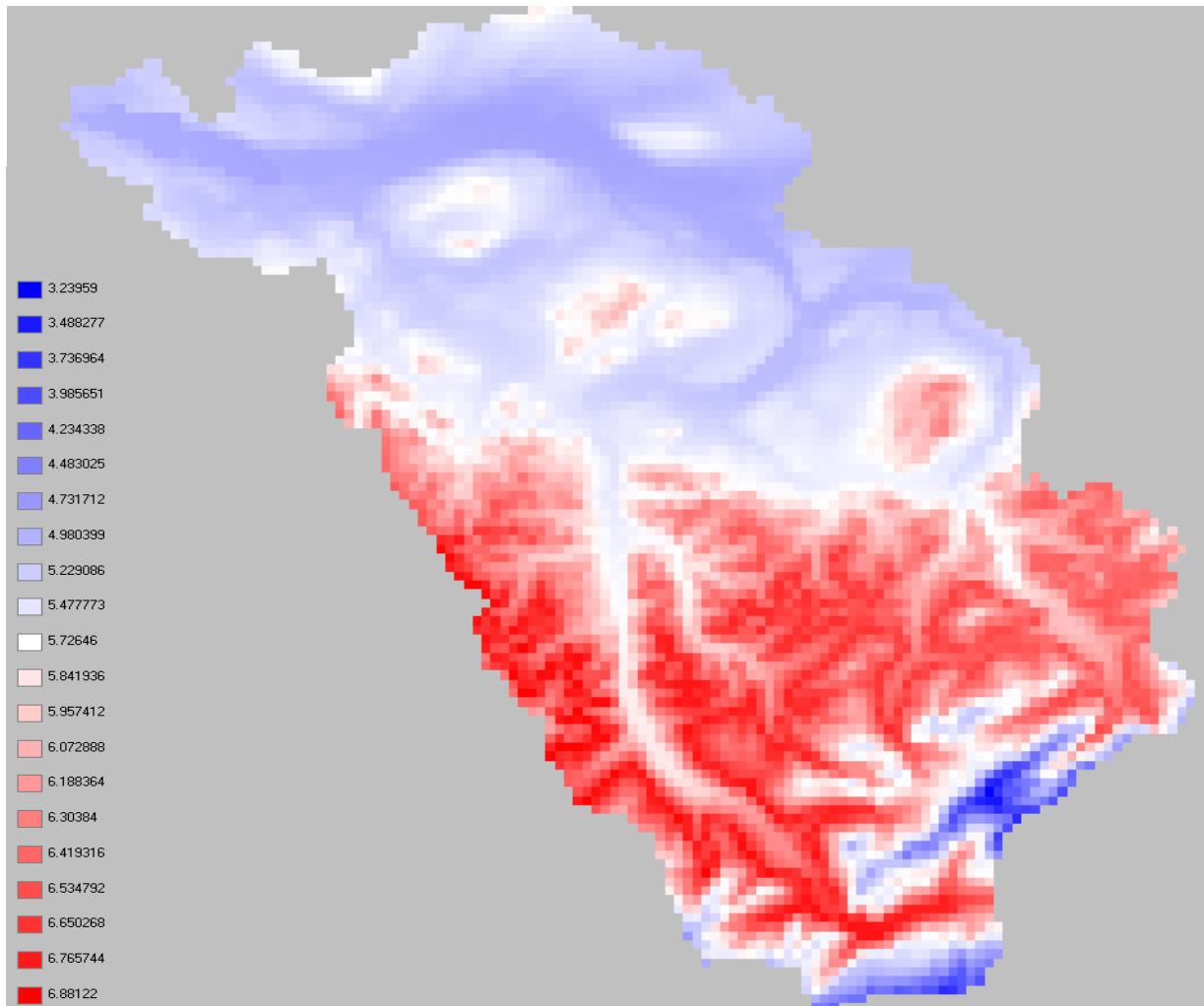


*Figure 1.2.4: Example for elevation dependent regression (January, 1, 1984, 14:00)*

For the lower left hand graph in figure 1.2.1, the according parameter set looks like this:

```
yy    mm dd hh   a1       b1       hv,l    a2       b2       hv,u     a3
1996  5  6  24   71.615   0.00000   0.0     71.615   0.06025   871.5    184.546
 b3       u         v1                         v2
-0.06933  220.05305 -2.3768683312431E-0004 9.86306324557048E-0005
```

Here, the lower inversion $h_{v,l}$ as well as the slope of the lowest regression $b_l$ is set to 0 to indicate, that only two regressions should be used. This mechanism may be problematic in regions with elevations < 0. The example above results in the following calculation for the interpolated value (same location as in the example before; here, the uppermost regression with index 3 is used):

$$T = \hat{u} + \hat{v}_1 x + \hat{v}_2 y - \hat{a}_1 + \hat{a}_3 + \hat{b}_3 z$$
$$= 220.053 - 0.0002377 \cdot 731000 + 0.0000986 \cdot 248000 - 71.615 + 184.546 - 0.06933 \cdot 1000$$
$$= 114.348 = 11.43°C$$

### 1.2.3 Linear combination of IDW + EDR (method 3)

Method ID for all methods using this algorithm: 3, and 11

This method combines methods 1 and 2 (using external preprocessing) and methods 1 and 10 (using internal regression preprocessing) respectively.

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
3                        # method
$inpath//regen//$year//.dat AdditionalColumns=0   # station data
$inpath//regen//$year//.out  # regression data
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid               # writecode
0.1                      # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998                      # error value
$IDWweight       # weighting of the reciprocal distance for IDW
0.75             # for method 3: relative weight of IDW
$IDWmaxdist      # max. distance of stations
$Anisoslope      # slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie     # ratio short to long axis of anisotropy-ellipsis
0.1                 # lower limit of interpolation results
0                   # replace value for results below the lower limit
900                 # upper limit for interpolation results
900                 # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3        # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1        # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

**Examples:**

Figure 1.2.5 shows a typical examples of a combination of IDW and EDR interpolations for the same date (January 3, 1994, 17:00) as used in the axamples for IDW. The weights are 0.75 for IDW and 0.25 for EDR.
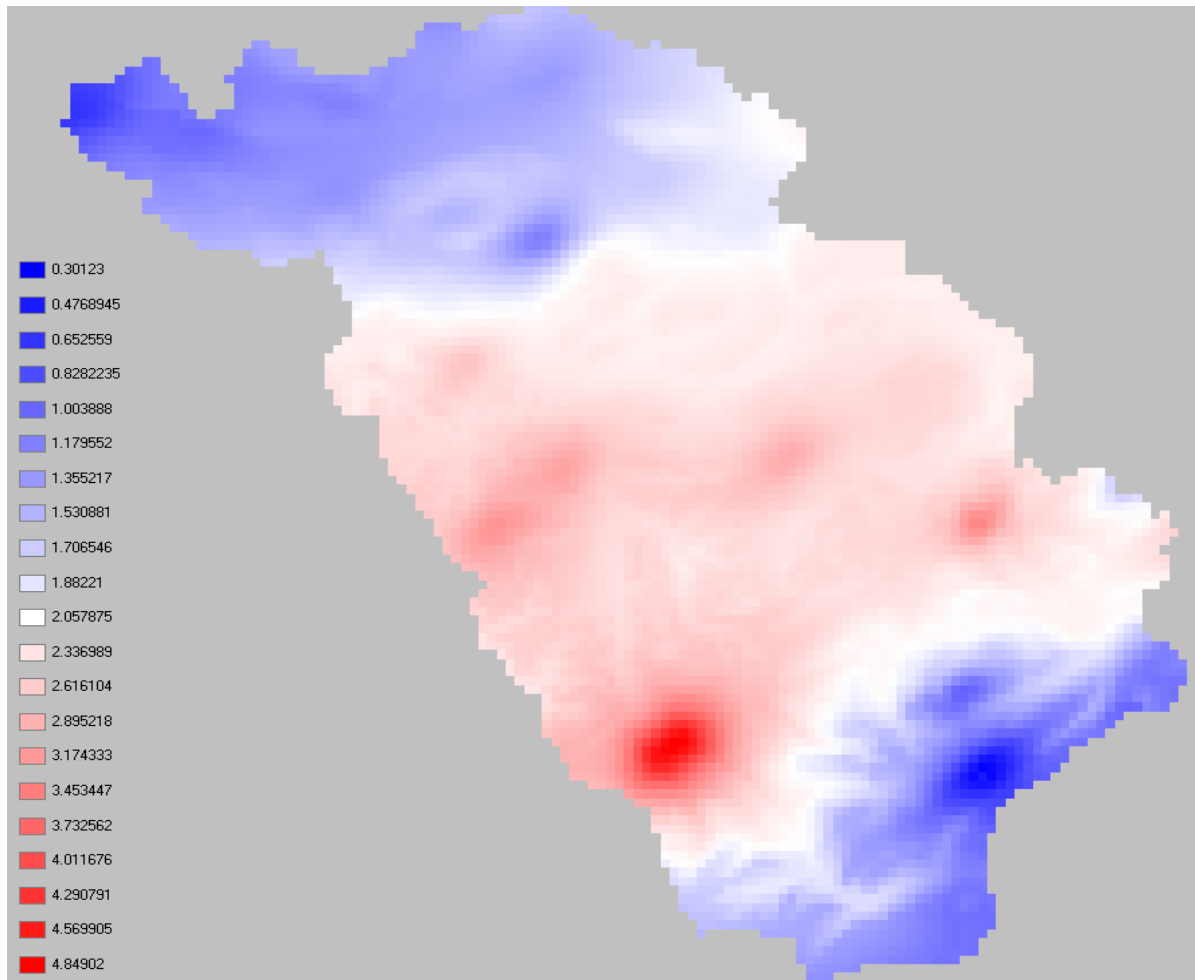
| | |
|---|---|
| ■ | 0.30123 |
| ■ | 0.4768945 |
| ■ | 0.652559 |
| ■ | 0.8282235 |
| ■ | 1.003888 |
| ■ | 1.179552 |
| ■ | 1.355217 |
| ■ | 1.530881 |
| ■ | 1.706546 |
| ■ | 1.88221 |
| ■ | 2.057875 |
| ■ | 2.336989 |
| ■ | 2.616104 |
| ■ | 2.895218 |
| ■ | 3.174333 |
| ■ | 3.453447 |
| ■ | 3.732562 |
| ■ | 4.011676 |
| ■ | 4.290791 |
| ■ | 4.569905 |
| ■ | 4.84902 |

*Figure 1.2.5: example for a combination of IDW + EDR (January 3, 1984, 17:00), same data as for examples for method 1*

### 1.2.4   Thiessen Polygons (method 4)

Thiessen poygons are the most simple method. It can be used if there is a single station only, but otherwise a more sophisticated method should be used. Each cell gets the value of the nearest station, thus creating a pattern of polygones delineating lines of equal distance to the two nearest stations.

Figure 1.2.5 shows an example for Thiessen polygones. The same data as for the examples in figures 1.2.1 and 1.2.2 are used.
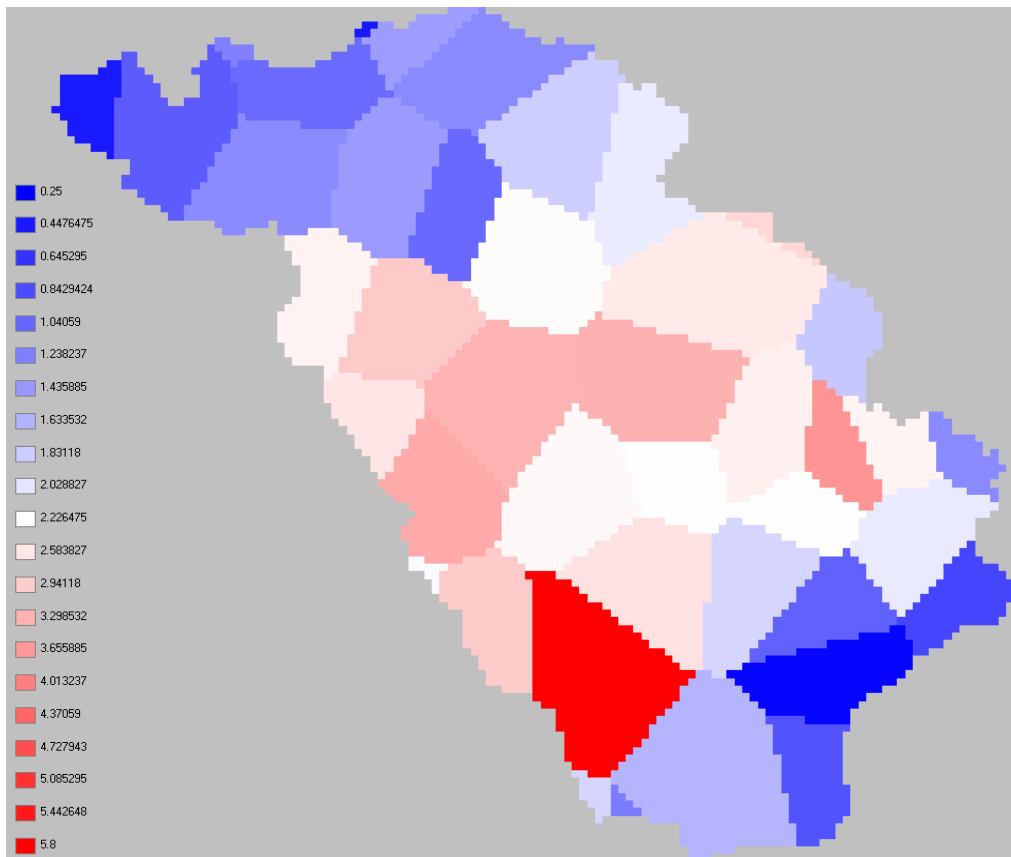
*Figure 1.2.6: example for interpolation using Thiessen polygones (precipitation, January 3, 1994, 17:00)*

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
4                            # method
$inpath//regen//$year//.dat AdditionalColumns=0   # station data
$inpath//regen//$year//.out  # regression data
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid                   # writecode
0.1                          # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998                          # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.75            # for method 3: relative weight of IDW
$IDWmaxdist     # max. distance of stations
$Anisoslope     #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie    #  ratio of short to long axis of the anisotropy-ellipsis
0.1                   # lower limit of interpolation results
0                     # replace value for results below the lower limit
900                   # upper limit for interpolation results
900                   # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3       # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1       # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.5 Bilinear interpolation (method 5)

Bilinear interpolation is particularly useful for downscaling meteorological input data which are already gridded, e.g. in case of using outputs of global circulation models (GCM) or weather models to drive WaSiM-ETH. However, the gridded input data are not read in as grids but as usual station data files as described in chapter 3.2.2, representing the GCM grid cells by separate columns which contain the "station" coordinates ($x$, $y$, $z$) in the header and one date per row for each model time step. Bilinear interpolation can also be used for usual climate station files, if the station positions are very close to a regular grid. Otherwise, artifacts may disturb a smooth interpolation surface.

Whilst usual bilinear interpolation requires the input data in a regular grid with equal spaces between the grid nodes all over the grid covering the entire interpolation area, this constraint was slightly liberated in WaSiM-ETH: At initialization, for each grid cell the next input station is looked for in each of the four quadrants. The station indexes are stored in four internal index-grids to speed up the later interpolation by avoiding wasting time by searching the stations again and again in every time step. By looking for the four nearest stations in the respective quadrants, the stations does not have to be arranged in a grid order at all. However, the bilinear interpolation assumes a more or less regular order, otherwise the interpolation results may show abrupt steps – but in principal it is possible to use any pattern of stations for this interpolation scheme. This liberation is also reasonable since GCM results are usually not given in regular grids but in latitude/longitude resolutions, so the distances between stations depend on the geographical location within the grid. If a station does not have four stations for interpolation available, i.e. if there is no station in at least one of the four quadrants or the actual data value is a nodata value, the interpolation method used for this cell is switched automatically to inverse distance weighting interpolation.
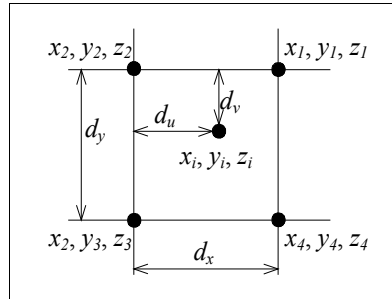


*Figure 1.2.7: scheme of calculating weights for bilinear interpolation to point i,j*

Figure 1.2.7 shows the scheme of calculating the weights for the interpolation location i,j. The weights are given by:

$$u = \frac{d_u}{d_x} = \frac{x_i - x_2}{x_1 - x_2} \quad \text{and} \quad v = \frac{d_v}{d_y} = \frac{y_i - y_2}{y_3 - y_2} \qquad (1.2.10)$$

with  $u$     weight of data of stations 1 and 4 (first and fourth quadrant) in $x$-direction
      $v$     weight of data of stations 3 and 4 (third and fourth quadrant) in $y$-direction
      $d_u$    distance of the interpolation location from $x_2$ in $x$-direction
      $d_v$    distance of the interpolation location from $y_2$ in $y$-direction
      $x_1,x_2$  $x$-coordinates of „stations" 1 and 2 (in quadrants 1 and 2, resp.)
      $y_2,y_3$  $y$-coordinates of „stations" 2 and 3 (in quadrants 2 and 3, resp.)

The interpolation result is then calculated by:

$$z_{i,j} = u \cdot (1-v) \cdot z_1 + (1-u) \cdot (1-v) \cdot z_2 + (1-u) \cdot v \cdot z_3 + u \cdot v \cdot z_4 \qquad (1.2.11)$$

To tell WaSiM-ETH that bilinear interpolation should be used, simply the interpolation method has to changed to 5 in the interpolation section of the respective climate variable.

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
5                              # method
$inpath//regen//$year//.dat AdditionalColumns=0   # station data
$inpath//regen//$year//.out  # regression data
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid                   # writecode
0.1                          # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998                          # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.75              # for method 3: relative weight of IDW
$IDWmaxdist     # max. distance of stations
$Anisoslope     #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie    #  ratio of short to long axis of the anisotropy-ellipsis
0.1                  # lower limit of interpolation results
0                    # replace value for results below the lower limit
900                  # upper limit for interpolation results
900                  # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3        # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1        # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.6 Linear combination of bilinearly interpolated gradients and residuals (method 6)

When interpolating e.g. the temperature or other altitude dependend variables, it may be important to consider small scale local variations in the digital elevation model. Therefor, another interpolation was introduced into WaSiM-ETH, the so called **BIGRES**-method (**B**ilinear **I**nterpolation of **G**radients and **RES**iduals). In fact, it is a linear combination of two independently performed biliniear interpolations, the first one interpolating the gradients, the second one interpolating the residuals. Both input data have to be read in as usual "station"-files, i.e., the gradients and residuals have to be calculated externally.

The results of both interpolations are then linearly combined using the digital elevation model as scaling parameter for the gradients and the residuals as offset:

$$z_{i,j} = zr_{i,j} + h_{i,j} \cdot zg_{i,j} \tag{1.2.12}$$

with  $z_{i,j}$      result of the linear combination
$zr_{i,j}$      result of the bilinear interpolation of the residuals (after 9.2.9)
$zg_{i,j}$      result of the bilinear interpolation of the gradients (after 9.2.9)
$h_{i,j}$      elevation from the digital elevation model [m]

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

To use this interpolation method, the method parameter in the interpolation section has to be set to 6. The residuals are given as the first file (the usal station file name). Then, and this is important, the 2nd filename for the gradients follows as an additional filename. The filename(s) of the regression file(s) must appear as usual! The following example shows a sample sequence from a control file using this interpolation method:

```
[temperature]
6                        # method
$inpath//t_0_100.dat AdditionalColumns=0   # station data
$inpath//grad100.dat AdditionalColumns=0   # station data
$inpath//temperature.out  # regression data
$outpath//tempgrid        # name of the output grid
$Writegrid                # writecode
0.1                       # correction faktor for results
$outpath//temp//$grid//.//$code//$year $hour_mean  # statistics file
998                       # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.75              # for method 3: relative weight of IDW
$IDWmaxdist     # max. distance of stations
$Anisoslope     #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie    #  ratio of short to long axis of the anisotropy-ellipsis
0.1                # lower limit of interpolation results
0                  # replace value for results below the lower limit
900                # upper limit for interpolation results
900                # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3       # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1       # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.7 Bicubic spline interpolation (method 7)

As an extension to the bilinear interpolation, also bicubic spline interpolation is applicable. The required input data are the same as for bilinear interpolation. The interpolation is done in three steps:

1. The stations are automatically organized in a grid-like structure (like for bilinear interpolation) including a virtual rotation of the domain

2. The north-south-meridians are spline-interpolated to the required density of information

3. An east-west-interpolation is done using the interpolated data of the meridians

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

To use this interpolation-method, the control file must be manipulated as shown for bilinear interpolation, except that the interpolation code is 7.

```
[temperature]
7                         # method
$inpath//t_0_100.dat AdditionalColumns=0   # station data
$inpath//temperature.out  # regression data
$outpath//tempgrid        # name of the output grid
$Writegrid                # writecode
0.1                       # correction faktor for results
$outpath//temp//$grid//.//$code//$year $hour_mean  # statistics file
998                       # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.75              # for method 3: relative weight of IDW
$IDWmaxdist     # max. distance of stations
$Anisoslope     #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie    #  ratio of short to long axis of the anisotropy-ellipsis
0.1                # lower limit of interpolation results
0                  # replace value for results below the lower limit
900                # upper limit for interpolation results
900                # replace value
```

```
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3        # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1        # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.8 Linear combination of spline interpolated gradients and residuals (method 8)

It's also possible to use a combination of bicubic-spline-interpolated resicuals and gradients, like for bilinear interpolation. The application is also identical like those, except that the interpolation code is 8 instead of 6.

The application of bicubic spline interpolation generally gives smoother surfaces, but it may happen, that unexpected results are shown: Since the interpolation makes a smooth curve ist possible, that minimum or maximum values exceed the value ranges of an entity. In this case the maximum and minimum parameter has to be set to proper values (e.g. for precipitation the minimum has to be set to 0, for humidity the minimum has to be set to 0, the maximum to 1).

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

To use this interpolation method, the method parameter in the interpolation section has to be set to 6. The residuals are given as the first file (the usal station file name). Then, and this is important, the $2^{nd}$ filename for the gradients follows as an additional filename. The filename(s) of the regression file(s) must appear as usual! The following example shows a sample sequence from a control file using this interpolation method:

```
[temperature]
8                             # method: combination of splines for
# residuals and gradients
$inpath//t_0_100.dat AdditionalColumns=0   # station data
$inpath//grad100.dat AdditionalColumns=0   # station data
$inpath//temperature.out  # regression data
$outpath//tempgrid        # name of the output grid
$Writegrid                # writecode
0.1                       # correction faktor for results
$outpath//temp//$grid//.//$code//$year $hour_mean  # statistics file
998                       # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.75              # for method 3: relative weight of IDW
$IDWmaxdist     # max. distance of stations
$Anisoslope     #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie    #  ratio of short to long axis of the anisotropy-ellipsis
0.1               # lower limit of interpolation results
0                 # replace value for results below the lower limit
900               # upper limit for interpolation results
900               # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3        # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1        # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.9 Reading externally interpolated data

If the input data are already available as interpolated grids, it is possible to configure WaSiM in a way that for each time step one of these grids is read instead of interpolating the data directly. Thus, outputs from GCM or RCM can be used with external interpolation methods. The only requirement is, that each grid is compatible to the grid dimensions and coordinates of the actual model run (ncols, nrows, cellsize, xll_corner and yll_corner). The grids may be organized on the hard disk or in any directory structure in any way – each grid may be specified with its full path and name.

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
9                               # method
$inpath//precip_gridlist.dat AdditionalColumns=0 # WaSiM grid list
$inpath//precip//$year//.out  # regression data
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid                       # writecode
0.1                              # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998                              # error value
$IDWweight       # weighting of the reciprocal distance for IDW
0.75                # for method 3: relative weight of IDW
$IDWmaxdist      # max. distance of stations
$Anisoslope      #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie     #  ratio of short to long axis of the anisotropy-ellipsis
0.1                    # lower limit of interpolation results
0                      # replace value for results below the lower limit
900                    # upper limit for interpolation results
900                    # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3        # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1        # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

The format of the grid list file looks similar to the WaSiM table file, except that the (single) data column contains valid file names instead of station data. This is an example for the grid list file:

```
grid list preipitation
YY    MM    DD    HH    9999
YY    MM    DD    HH    9999
YY    MM    DD    HH    9999
YY    MM    DD    HH    Liste
1984  1     1     1     d:\Data\m500\input\meteogrids\precm500_0101.001
1984  1     1     2     d:\Data\m500\input\meteogrids\precm500_0101.002
1984  1     1     3     d:\Data\m500\input\meteogrids\precm500_0101.003
1984  1     1     4     d:\Data\m500\input\meteogrids\precm500_0101.004
1984  1     1     5     d:\Data\m500\input\meteogrids\precm500_0101.005
1984  1     1     6     d:\Data\m500\input\meteogrids\precm500_0101.006
1984  1     1     7     d:\Data\m500\input\meteogrids\precm500_0101.007
1984  1     1     8     d:\Data\m500\input\meteogrids\precm500_0101.008
1984  1     1     9     d:\Data\m500\input\meteogrids\precm500_0101.009
1984  1     1     10    d:\Data\m500\input\meteogrids\precm500_0101.010
```

### 1.2.10 Elevation dependent Regression with internal preprocessing (EDRINT)

This method was described in details in chapter 1.2.2 already. This is a reference to this section only.

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
10                              # method
$inpath//regen//$year//.dat AdditionalColumns=0    # station data
$inpath//temper//$year//.out # file name with regression data
$inpath//regen//$year//.out  # regression data
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid                      # writecode
0.1                             # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998                             # error value
$IDWweight      # weighting of the reciprocal distance for IDW
0.75            # for method 3: relative weight of IDW
$IDWmaxdist     # max. distance of stations
$Anisoslope     #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie    #  ratio of short to long axis of the anisotropy-ellipsis
0.1                   # lower limit of interpolation results
0                     # replace value for results below the lower limit
900                   # upper limit for interpolation results
900                   # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3       # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1       # number of scenario cells
699000 235000    0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.11 linear combination of IDW and EDRINT (method 11)

This method was already described in detail in chapter 1.2.3. This is a reference to this section only.

**Practical application:**

**Control file:** (red, bold and large font: specific settings for method IDW)

```
[precipitation]
11                              # method
$inpath//regen//$year//.dat AdditionalColumns=0    # station data
820 1400 200 1 300 # lower inversion [m asl], upper inversion
#  [m  asl],  tolerance  [m],  overlap  [0/1  for  true/false],
#clusterlimit [m]
$inpath//regen//$year//.out  # regression data
$outpath//idw_//$preci_grid  # name of the output grid
$Writegrid                      # writecode
0.1                             # correction faktor for results
$outpath//idw_prec//$grid//.//$code//$year $hour_mean  # statistics file
998             # error value
$IDWweight          # weighting of the reciprocal distance for IDW
0.75                # for method 3: relative weight of IDW
$IDWmaxdist         # max. distance of stations
$Anisoslope         #  slope of the anisotropy-ellipsis (-90+ to +90°)
$Anisotropie        # ratio short to long axis of anisotropy-ellipsis
0.1                 # lower limit of interpolation results
```

```
0                     # replace value for results below the lower limit
900                   # upper limit for interpolation results
900                   # replace value
$SzenUse    # 1=use scenario data for correction, 0=dont use scenarios
3       # 1=add scenarios, 2=multiply scenarios, 3=percentual change
1       # number of scenario cells
699000 235000     0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
# coordinates of the cells, then one value for each month of a year
```

### 1.2.12 Using multiple interpolation methods at the same time (Regional Superposition)

The definition of interpolators is generally not limited by WaSiM-ETH. It is possible to define as many interpolation sections (each with its own identification string) as the hardware will be able to process. However, the internal identification strings are limited to the following list:

- global_radiation      used in evaporation and snow/glacier melt
- net_radiation      used in evaporation and snow/glacier melt
- sunshine_duration      used in evaporation
- temperature      used in evaporation, snow/glacier melt, permafrost thawing and dynamic phenology
- temperature_14      used for evaporation (Wendling)
- wind_speed      used for evaporation
- vapor_pressure      used for various evaporation approaches
- vapor_pressure_14      used for evaporation (Wendling)
- air_humidity      used for various evaporation approaches
- air_humidity_14      used for evaporation (Wendling)
- air_pressure      not used yet
- precipitation      input for interception/snowmelt/soil model etc.

Multiple interpolation methods for the same data type can be used in different ways:
1. Using two (or more) methods for the entire basin and superposing the results with specific weights for each method. Prior to Version 8.5 this was already possible with interpolation method 3 (and now also 11), but for IDW and EDR(EXT) only.
2. Using specific interpolation methods for different regions with a smooth transition. This was partly possible prior to version 8.5: Both methods must be EDR, but there could be as many as up to 30 regions using specific parameter files each. Starting with version 8.5, any of the interpolation methods described above may be used for any of the regions defined in the region grid
3. It is even possible to combine both uses: multiple interpolation methods can be used for each region with different weights, offering the greatest flexibility thinkable at this time.

A special feature is the smooth transition of the interpolation results between regions.

**Defining Regions:**

The regions are given as a grid with the same geometry as the zone grid (same numbers of rows and columns, same cellsize, nodata value, lower left x- and y-coordinates). Each region is identified by a unique code. The codes should be integer numbers of any order, no ranking is required, gaps are allowed (e.g. if using codes 4, 1 and 8 following any special notation).

The regions are technically totally independend on the zones or subbasins, except for the fact, that each valid grid cell of the zone grid must belong to one of the regions. Zones are used for the statistic output of the model results whereas regions are used for selecting the matching interpolator.

Near the borders between the regions the model produces a smooth transition by weighing the regression interpolation results of up to three regions dependent on the distances of the actual cell to the respective regions.

If there are multiple regions defined, the model performs the following tasks:

Tasks at initialization:

- Reading the region grid (the identification string for the region grid in the control file has to be "regression_regions" in lower cases without quotes).

- Reading the transition distance, i.e. the maximum distance of a cell to any regions border to be affected also by the neighboring region(s). This parameter is given in meters in the new section [region_transition_distance] of the control file.

- Generating a REGION2- and a REGION3-grid. The REGION2 grid contains for each grid cell the region code of the second region the cell is affected by. If the grid cell is not within the maximum transition distance from any region border, than the REGION2 grid contains a zero-value in the matching location. In Analogy to REGION2, the REGION3 grid contains the codes of the third region the cell is affected by, also only if there is a third affecting region at all. The REGION2 and REGION3 grids are written into the directory the region grid was read from with the file extensions „.$r2" and „.$r3", respectively, for the modellers reference.

- Generating a WEIGHT1, WEIGHT2 and WEIGHT3-grid containing the weights the respective regions have for the interpolation result. When doing the interpolation, for each grid cell the up to three region grids are checked. If the matching positions contain codes greater than zero, the actual result is scaled with the weight of the according WEIGHT<n> grid. The weights are calculated after the following scheme. If a cell in row $i$ and column $j$ is affected by only one region (coded in the grid REGION1), the weights are $\text{WEIGHT1}_{i,j} = 1.0$, $\text{WEIGHT2}_{i,j} = 0$ and $\text{WEIGHT3}_{i,j} = 0$ for the regions indexed in the matching grids $\text{REGION1}_{i,j}$, $\text{REGION2}_{i,j}$, and $\text{REGION3}_{i,j}$, respectively.

*region weights for cells affected by two regions*

Weights for cells affected by two regions are calculated using a linear transition starting with a weight of 1.0 if the cell is at the maximum transition distance from the region border and decreasing to 0.5 if the cell is directly at the border. This weight is stored in $\text{WEIGHT1}_{i,j}$, the weight of the neighboring regions regression result is then simply $\text{WEIGHT2}_{i,j} = 1.0 - \text{WEIGHT1}_{i,j}$.

$$w1_{i,j} = 0.5 + 0.5 \cdot d/r$$
$$w2_{i,j} = 1.0 - w1_{i,j}$$

*(if only 2 regions affect the actual cell)* **(1.2.13)**

with  $w1_{i,j}$   weight of the regression result using the parameters for the region the cell belongs to

$w2_{i,j}$   weight of the regression result using the parameters for the second region the actual cell is affected by

$d$   distance of the actual cell to the region border [m]

$r$   maximum distance to be considered for smooth transitions, specified in the control file in the section [region_transition_distance], unit [m]

*region weights for cells affected by three or more regions*

If a cell is affected by three ore more regions, only the three nearest regions are used for calculating weights. The sketch in figure 1.2.8 illustrates the calculation of weights. The maximum distance from any region border a point must have to be not influenced by another region is the radius $r$.
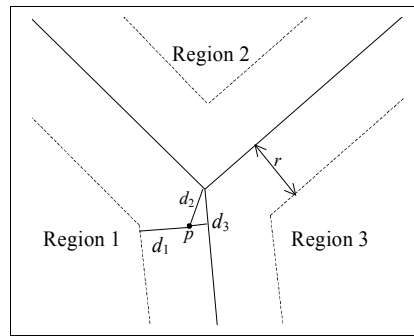
*figure 1.2.8: Weighting of multiple regions for a point p near the regions crossing point*

If the interpolation point is only affected by one neighboring region, the above formula (1.2.13) is applied. Only if the crossing point of three regions is within the distance $r$ around the interpolation point, two neighboring regions are considered for calculating the weights. The weights are calculated by a modified inverse distance weighting scheme. The following constraints must hold:

➢ (1)    Regardless of the shape of the region borders, a point at the crossing point of three regions have to have a weight of 1/3.

➢ (2)    The weight of a region decreases with increasing distance $d$ of the point to that region until it reaches the value of 0.0 at the distance $d = r$.

➢ (3)    The rate of increase of the weights should be linearly related to the decreasing distance to the nearest region in order to match the linear transition of weights between two regions at the outer limit of the three-region-affected area.

➢ (4)    The sum of all three weights must be 1.0.

In a first step the distances $d_2$ and $d_3$ are extended by $r$ in order to satisfy condition (1) resulting in:

$$
\begin{aligned}
u_1 &= d_1 \\
u_2 &= r + d_2 \\
u_3 &= r + d_3
\end{aligned}
\qquad\qquad\text{(1.2.14)}
$$

If $d_2 = d_3 = 0$, then $u_1 = u_2 = u_3 = r$ resulting in equal weights of 1/3 for all regions. To satisfy conditions (2) and (3), the inverted distances $u_1$ to $u_3$ are modified in the following way:

$$
\begin{aligned}
x_1 &= \frac{1}{u_1} - c \cdot \frac{1}{2r} \\
x_2 &= \frac{1}{u_2} - c \cdot \frac{1}{2r} \\
x_3 &= \frac{1}{u_3} - c \cdot \frac{1}{2r}
\end{aligned}
\qquad\qquad\text{(1.2.15)}
$$

with  $x_1 ... x_3$    modified inverse distances from the interpolation point to the three regions of interest

   $u_1 ... u_3$    distances $d_1$ to $d_3$ to the three regions after figure 1.2.8, extended by $r$ after equation (1.2.14).

   $r$    maximum distance of the interpolation point to another region to be affected by this region

   $c$    correction factor for modifying the inverse distances

The correction factor $c$ depends on the minimum distance of the interpolation point to any

regions interior area (usually of region 1) and is responsible for satisfying condition (3). It is estimated after:

$$c = 0.9 - \frac{1}{2} \cdot \cos\left(\frac{\pi}{2} \cdot \frac{\min(u_1, u_2, u_3)}{r}\right)$$
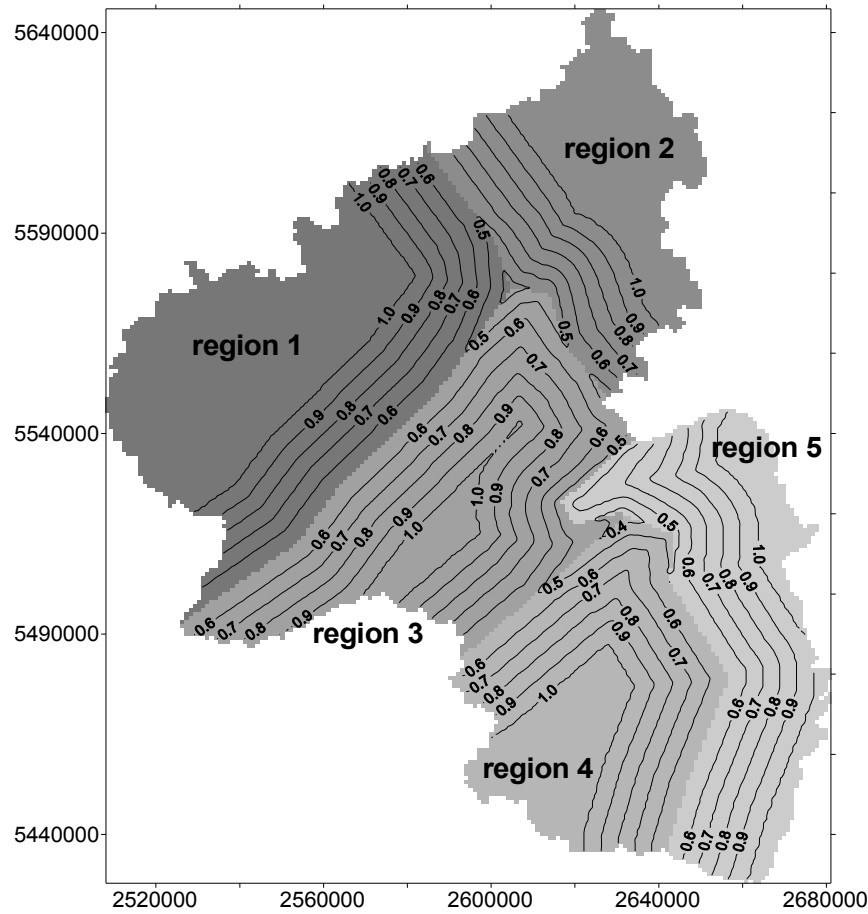
(1.2.16)



*figure 1.2.9: Weighting of the actual region if using multiple regions. Rheinland-Pfalz, Germany, 5 regions, maximum transition distance r was set to 20 km.*

The weights of the regions are then calculated by the usual inverse distance weighting scheme:

$$w_i = \frac{1}{d_i} \cdot \frac{1}{\sum\limits_{i=1}^{3} d_i} \qquad (i=1..3)$$

(1.2.17)

with $w_i$    weighting of the regions 1...3 for the modified inverse distance interpolation

Figure 1.2.9 shows the weights for the regression parameter sets of the regions 1 to 5 for cells which belong to the respective regions. Such weighting grids are also generated for the second and third region affecting the actual regions. The sum of all three weighting grids is 1.0 for all valid region cells.

Tasks at interpolation:

- Each time the interpolation is done, the region grids REGION1$_{i,j}$, REGION2$_{i,j}$, and REGION3$_{i,j}$ are checked for entries unequal to zero for each location $i,j$. Region grid REGION1 (the only grid the model reads in at initialization time) must contain valid entries for each grid cell which is valid in the zone grid, whereas the other region grids (which are internally created during initialization) contain only valid entries, if the cell is affected by other regions when its location is within the maximum transition distance between two or more regions $r$.

- calculating the interpolation results by weighing the results of the at maximum 3 affecting regions. **Note: each single result may already be the result of a superposition of multiple interpolation methods. Here, only the generation of a smooth transition between regions is described. The superposition itself is described later in this chapter.**

- combining the results by using the region weights for the actual location:

$$z = w_1 z_1 + w_2 z_2 + w_3 z_3 \qquad\qquad (1.2.18)$$

with $z$       weighted result of altitude dependend regression for the actual location

$z_1...z_3$     results of the altitude dependend regressions for the three regions affecting the actual location

$w_1...w_3$   weights for the three regions after equation (1.2.17)

**Defining rules for weighted superposition:**

- The region grid must be read in as a standard grid, it may have any file name, but the identification string must be *regression_regions*, the fill parameter must be 1:

```
[standard_grids]
...
$inpath//$grid.reg        regression_regions 1     # regression regions
...
```

- The section [region_transition_distance] must be created in the control file containing the maximum distance, a cell can be located from any other region to be affected by this other region:

```
[region_transition_distance]
10000 # maximum transition distance in meter
```

- The control file must contain a new section [RegionalSuperposition]. An example is given below. This section contains the main switch (0 or 1), the time step, the numbers of superpositions to be carried out and then for each of the superpositions a description starting with the resulting entity name (e.g. precipitation). Each input from an interpolator must be identified by the name given to this interpolator (e.g. precipitation_reg) and a relative weight must be defined for each region (when no regions are used, then default region 1 must be attached a weight of 1.0). After the list of entityinputgrids the name of the outputgrid and the statistics file with its common writecodes must be given.

```
[RegionalSuperposition]
1
$time
NumberOfEntities = 2;
precipitation {
        entityinputgrid = precipitation_reg ;
                regions = 1    2    ;
                weights = 0.0 0.2 ;
        entityinputgrid = precipitation_idw ;
                regions = 1    2    ;
                weights = 1.0 0.8 ;
        outputgrid  = $outpath//$preci_grid ;
                writecode = $Writegrid ;
        outputtable = $outpath//prec//$grid//.//$code//$year;
                statcode  = $hour_mean;
}
temperature {
        entityinputgrid = temperature_reg;
                regions = 1    2    ;
                weights = 0.7 0.8 ;
        entityinputgrid = temperature_idw;
                regions = 1    2    ;
```

```
        weights = 0.3 0.2 ;
    outputgrid  = $outpath//$tempegrid ;
        writecode = $Writegrid ;
    outputtable = $outpath//temp//$grid//.//$code//$year ;
        statcode  = $hour_mean ;
}
```

**Note 1**: all weights must add up to 1.0 for each single region

**Note 2**: all regions must be listed in the regions list for each input grid. If a specific interpolation should not be considered for a region, then the weight must be set to 0.

**Note 3**: there may be as many interpolators combined as required. It's not limited to 2 methods only. Even the same method with different input data could be used (this is the equivalent to the old method for regression only).

**Note 4**: When using multiple regions with superpositions, precipitation correction will be carried out AFTER interpolation and superposition of interpolation results because it is not predictable before superposition runs what values temperature and wind speed will have at the stations location. Be aware of artifacts (steps) due to non-linearities in the precipitation results in case of temperatures in the range around the threshold for snow/rain. Since another correction for snow may apply, the resulting maps may show steps where the temperature suddenly falls below the snow/rain threshold (which was not the case when correcting the precipitation at station locations only). It's recommended to apply precipitation correction prior to interpolation, i.e. to feed WaSiM with already corrected precipitation.

**Note 5**: the separate interpolators should be named in a speaking manner, like precipitation_idw for an interpolator with method IDW and precipitation_reg for an interpolator with method regression. The definition of the superpositions will then define the name as expected in the model (see list with allowed names). Precipitation must then always be named as precipitation, temperature as temperature or temperature_14 etc. (the submodels expect the precipitation, temperature etc. to have exactly these names).

**Examples:**

Figure 1.2.10 shows the region grid for the river Thur basin (Switzerland). The northern part (in blue) is the lower region, the southern part (in red) it the mountainous region. Here, the regions are defined following sub basin borders (cellsize: 500m). However, this is not required. Regions may have any shape.

Figures 1.2.11 to 1.2.14 show the superposition results for temperature () with varying transition zones: 0km (no transition) for figure 1.2.11, 1km for figure 1.2.12, 5km for figure 1.2.13 and finally 10km for figure 1.2.14. The IDW and EDR interpolations were done for the entire basin. During the regional superposition, the local values of each interpolation method were weighted according to the description:

```
temperature {
    entityinputgrid = temperature_reg;
        regions = 1   2   ;
        weights = 0.0 1.0 ;
    entityinputgrid = temperature_idw;
        regions = 1   2   ;
        weights = 1.0 0.0 ;
    outputgrid  = $outpath//$tempegrid ;
```

```
          writecode = $Writegrid ;
      outputtable = $outpath//temp//$grid//.//$code//$year ;
          statcode  = $hour_mean ;
}
```

This means, that for the northern part Only IDW is used whereas for the southern part only EDR is used. The transition zone, however, uses the results of both methods according to the weights of the affected regions.
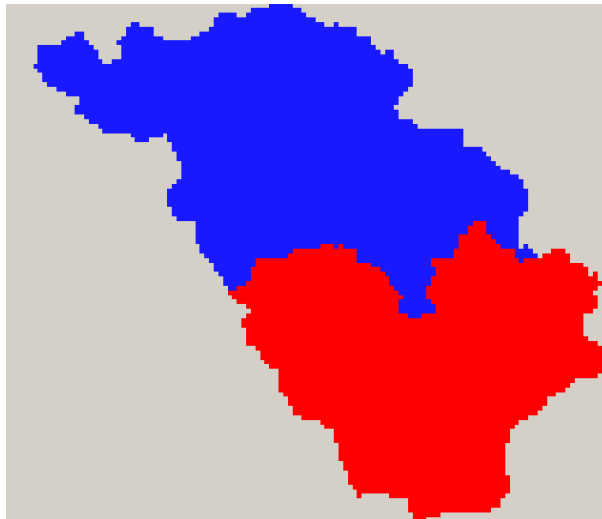


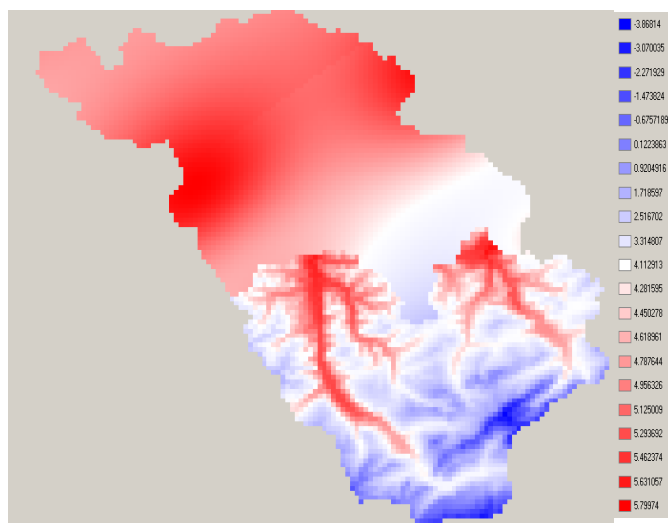*Figure 1.2.10: Definition of two regions for the river Thur basin (Switzerland, 1700km²)*



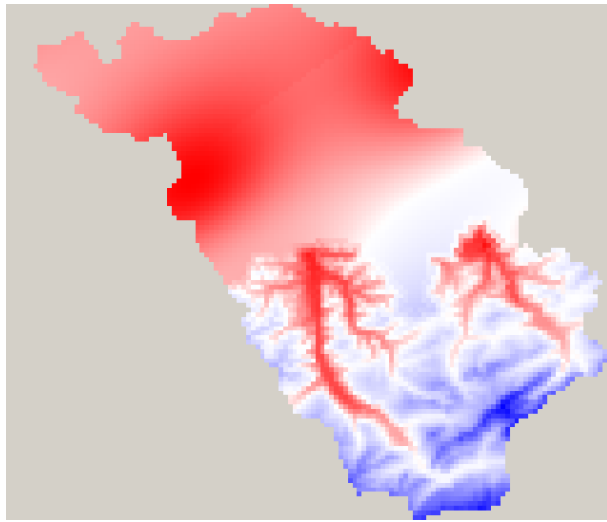*Figure 1.2.11:  interpolation results for IDW (North) and EDR (South) without smooth transition*

*Figure 1.2.12: interpolation results for IDW (North) and EDR (South) without 1km transition-range*
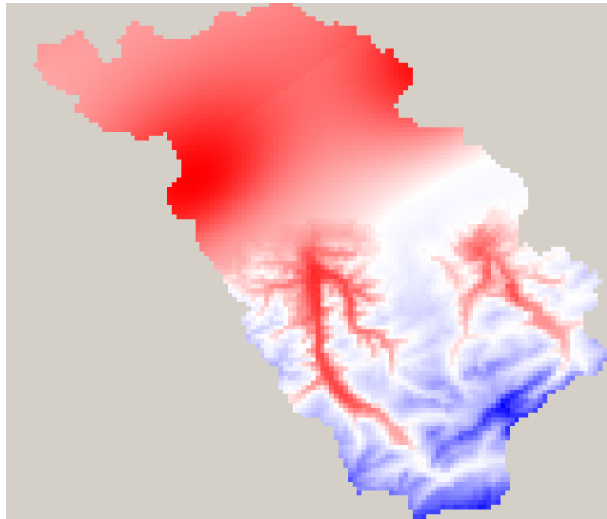


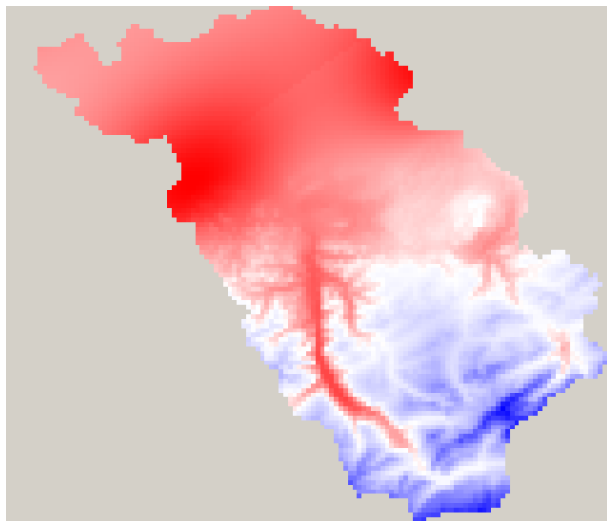*Figure 1.2.13: interpolation results for IDW (North) and EDR (South) without 5km transition-range*



*Figure 1.2.14: interpolation results for IDW (North) and EDR (South) without 10km transition-range*