Answer sheet: Spatial Multi Criteria Evaluation for qualitative risk assessment.

5.2 Generic social vulnerability indicators

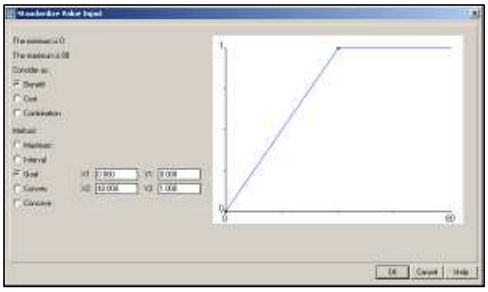
5.2.1. Problem definition:

QUESTION: Apart from the criteria that are given here, which other indicators do you think could be used in determining social vulnerability? Name a few examples, and indicate where you could get such data from, in your own country.

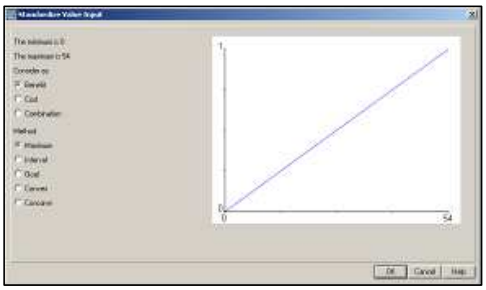
ANSWER: *Disabled people* (from the municipality or governmental authority who gives sanitary assistance), *family with more than 6 people* (from municipality or census), *female population* (from municipality).

5.2.2. Standardization of the factors

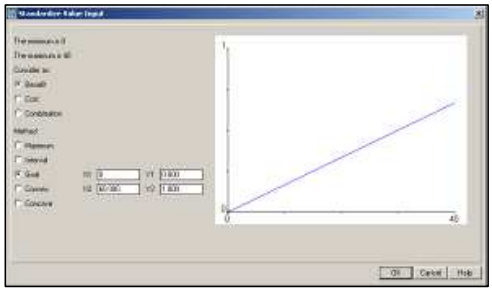
The images below show the standardization used on the exercise. As you can see has been used different methods. This is one of the possible standardization and probably your own evaluation is different.



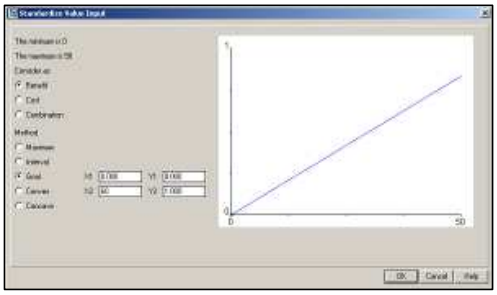
Under poverty level



Unemployment



Social structure related.

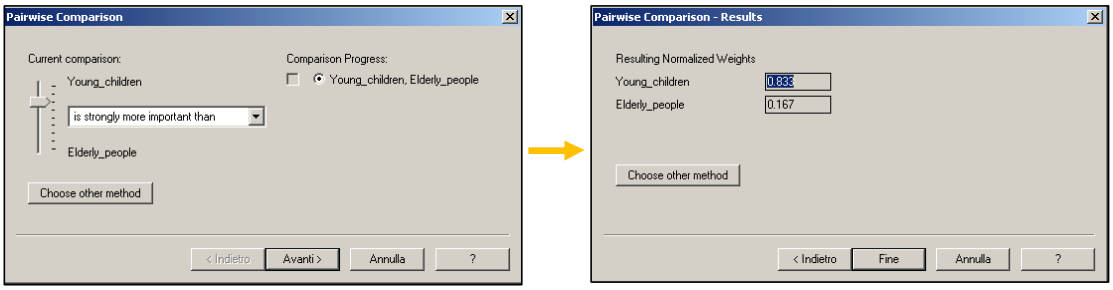


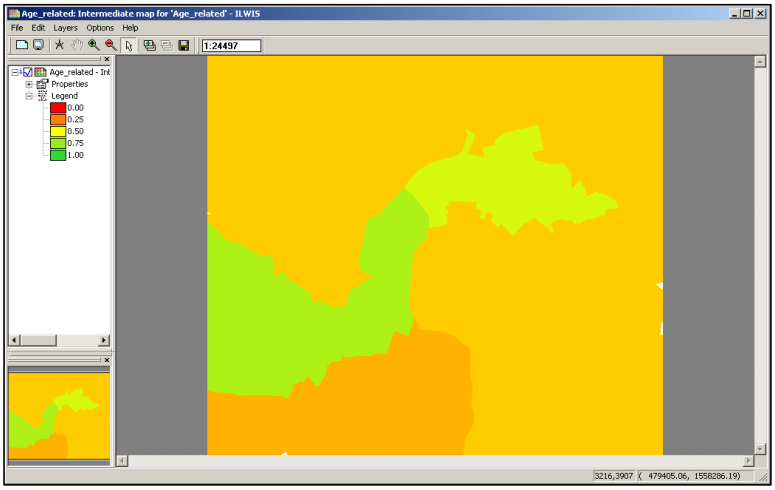
Single parent households

5.2.3. Determining the weights among factors

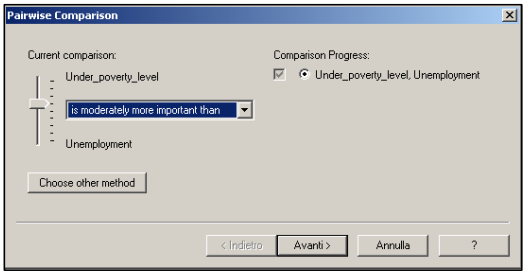
In order to determine the weight among the factors, using the pairwise method you can compare a couple of factor according to a qualitative classes of relevance and then check on the next step the quantitative evaluation derived from your choice. At the end is shown the Age_related_map resulted from the pairwise comparison.

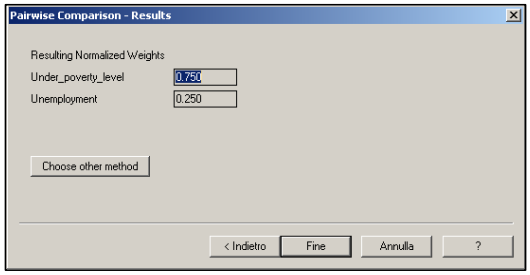
Age related:

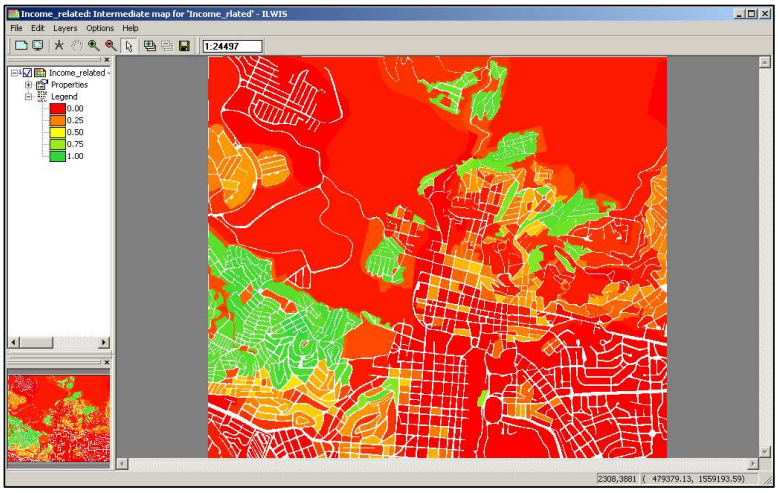




Income related:

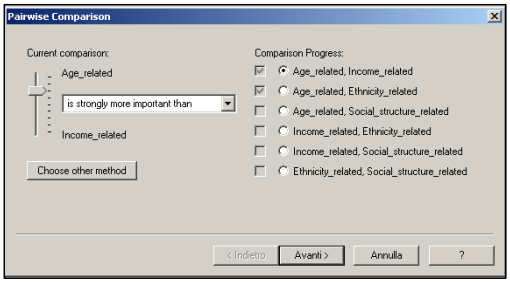


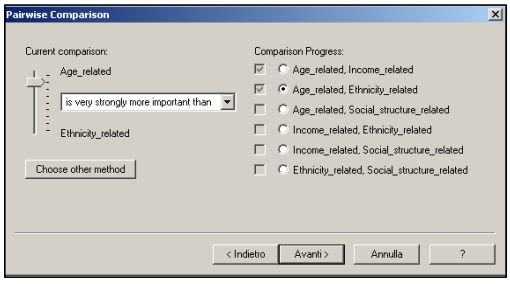


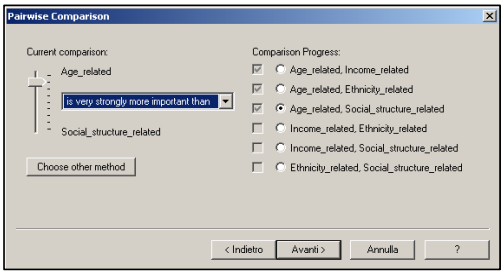


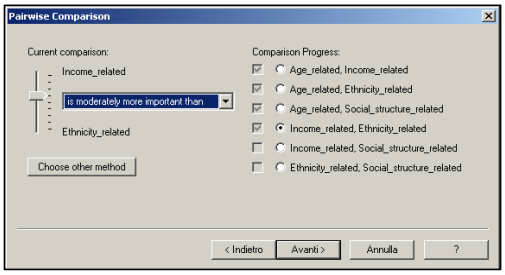
For the ethnicity related and structural related there is only one factors we are not showing here the result raster maps.

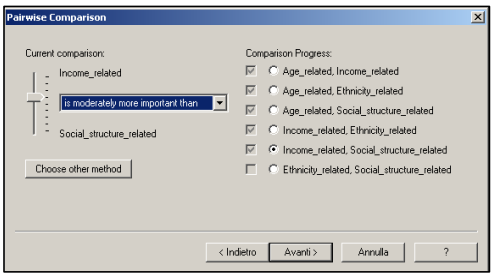
5.2.4 Determining the weights among groups

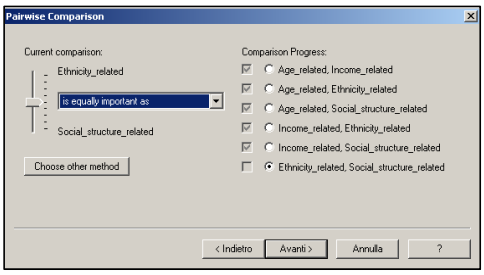


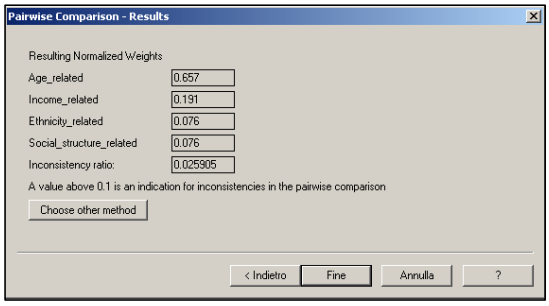








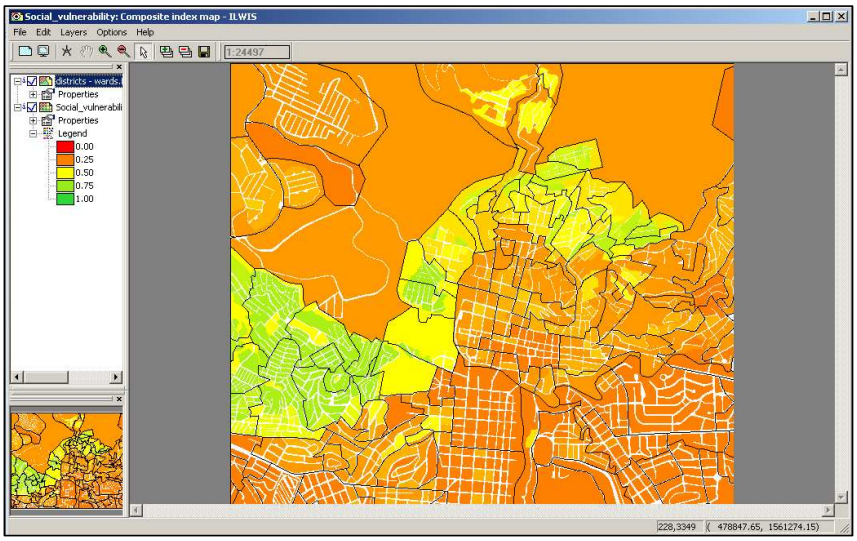




It is possible that in your comparison some factors are inconsistent. This much probable that could happens when in the tree there are several factors. For example if you select A > B > C, we should select even A > C otherwise will be an inconstancies.

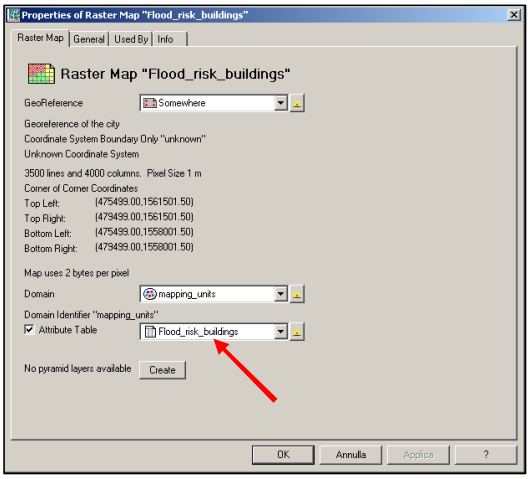
Comment [j1]: Check if necessary try to evaluate again without inconsistencies

from the pattern of the social vulnerability is possible see that the most vulnerable areas are in correspondence of the districts "**Europe**" and "**Australia**" with a maximum values in the landuse type **residential_squatter**.

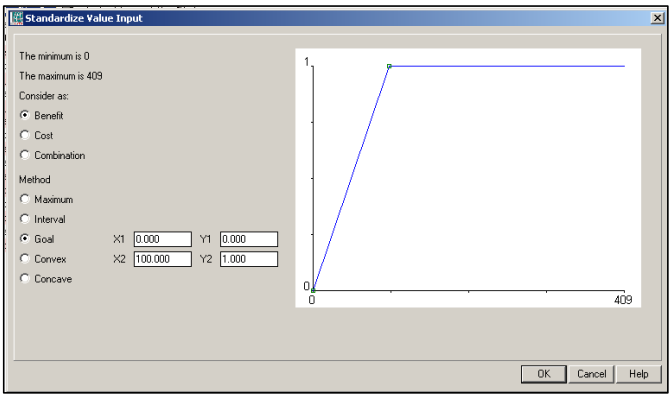


5.3 Hazard specific population vulnerability indicators

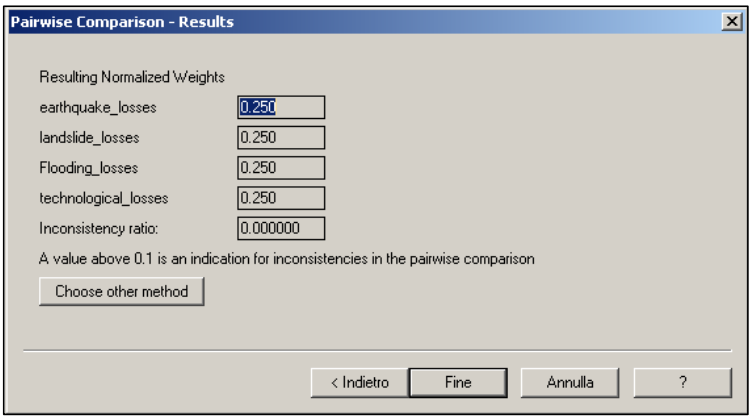
Below is shown the box of the properties of Flood_risk_buildings, where you can see that the attribute table linked has been changed.



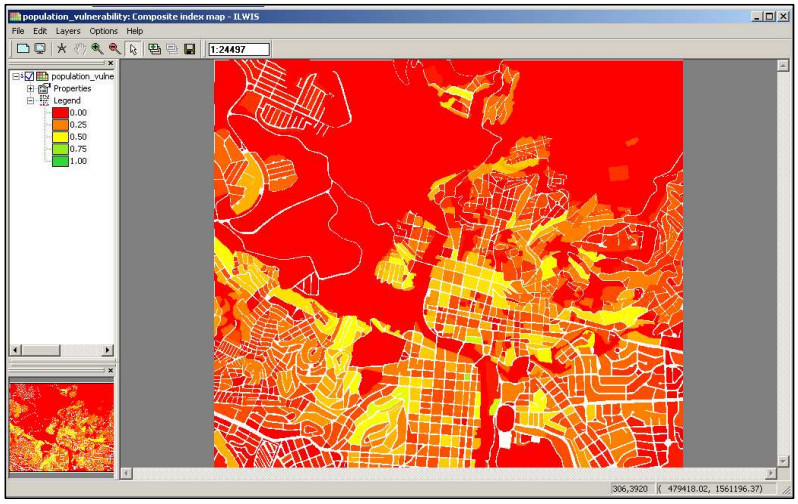
The following box shows the standardization and the values used for the indicators.



The weights among the groups have been considered as equal each other because it is possible to think to an equal importance to the losses becoming from different hazard type. In other word only the number of people affected (according to the standardization made) will determine the values of population vulnerability.



The population_vulnerability maps is shown below.



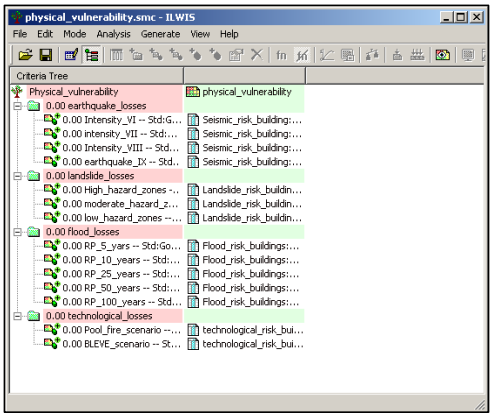
- Do you think that the parameters taken in account are good indicators for the evaluation of the vulnerability? Do you have other ideas?

Comment [j2]: Question and answer added by me. the question has been included on the text of the exercise.

The number of people affected per scenario is expressed per mapping units. This means that mapping units with same number of people but different dimension, will be considerate with the same vulnerability index. An alternative way could be consider the density of people affected per mapping units (dividing the number of people affected per the area of the mapping unit itself).

5.4 Hazard specific physical vulnerability indicators

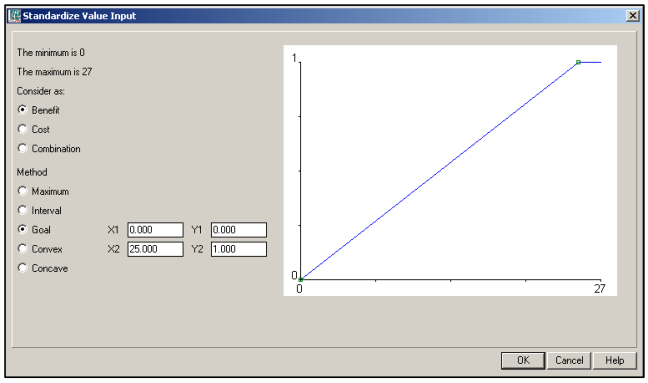
Generating the criteria tree



Standardizing and weighting

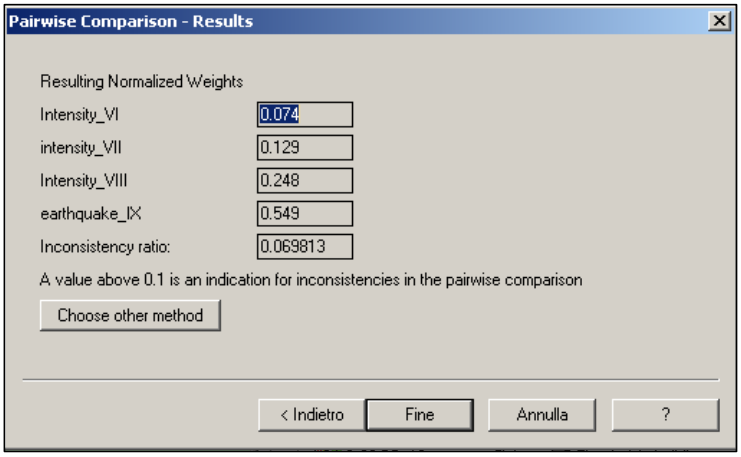
For the standardization has been used the goal standardization and the value of 25.

Comment [j3]: Maybe could be a fine check the maximum value of building losses in the area in the worse scenario and standardize using this value as maximum. In that way at all the losses will be assigned a value proportioned to the ratio of losses in the overall riskcity. Maybe if this observation is good could be added in a NOTE BOX.

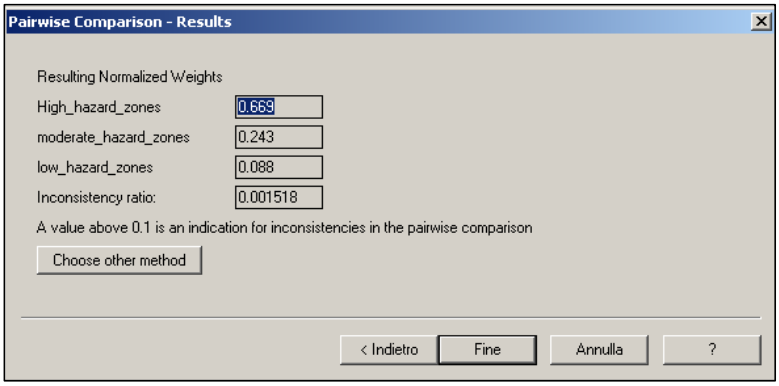


During the comparison among the factors, every scenario has been considered moderately less important than the subsequent higher level of intensity (for the case of earthquake).

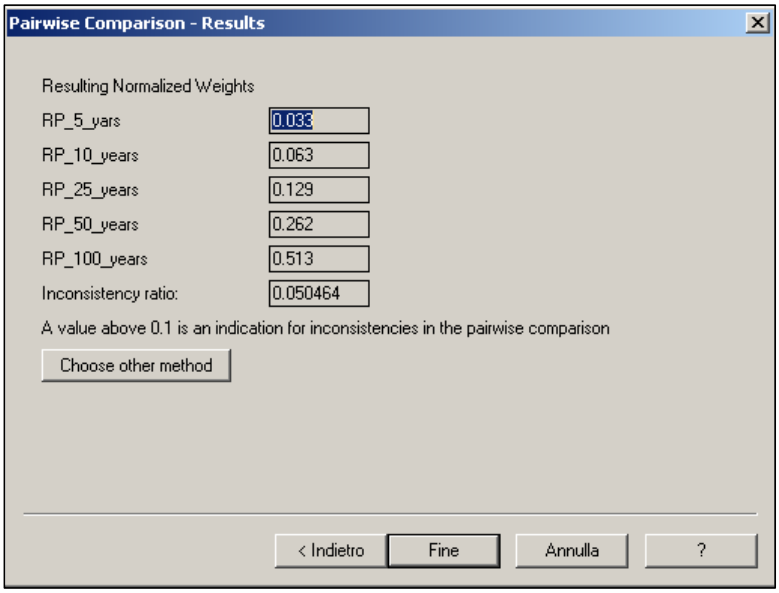
Comment [j4]: Is also possible consider the same weight for every scenario, since the importance of the losses (for the same losses values) should be the same, and not depending from the intensity. With the weight used on the exercise, I am accentuating the highest intensity.



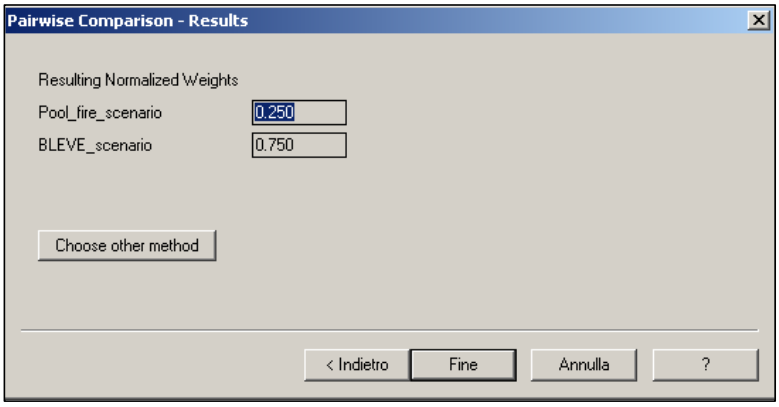
Earthquake Pairwaise comparison.



Landslide losses pairwise comparison

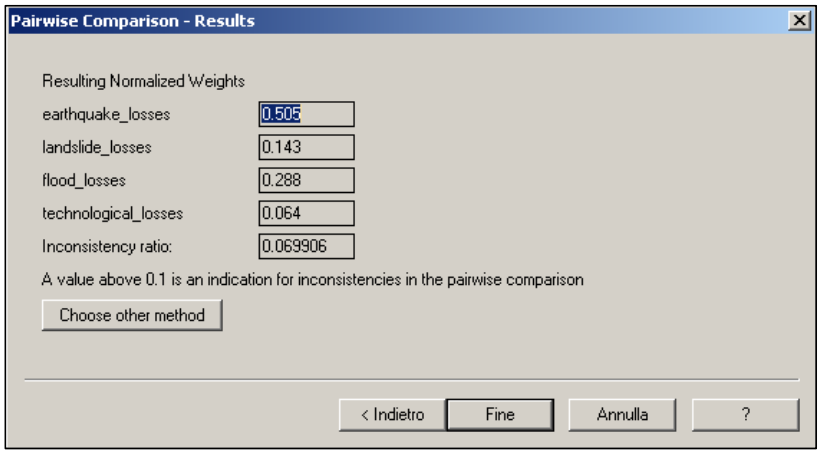


Flood losses pairwise comparison

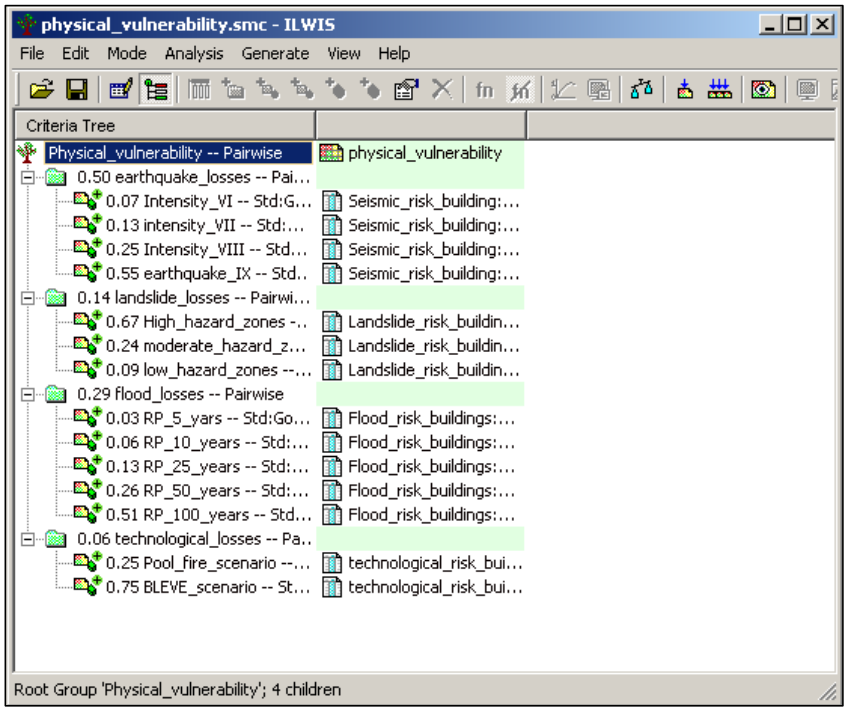


Technological losses pairwise comparison

Weighting among the groups.



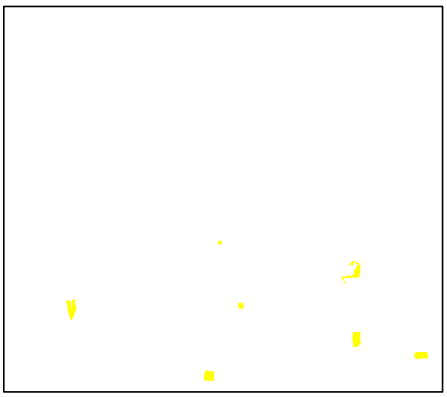
The physical vulnerability tree should looks like the image below (it is possible that your values are different)

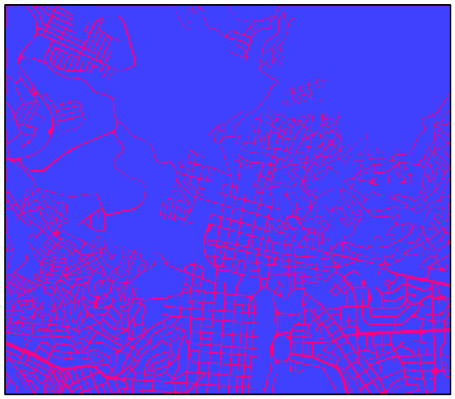


5.5 Capacity indicators

Below is shown the methods for the evaluation of the distance of every mapping units to the hospitals.

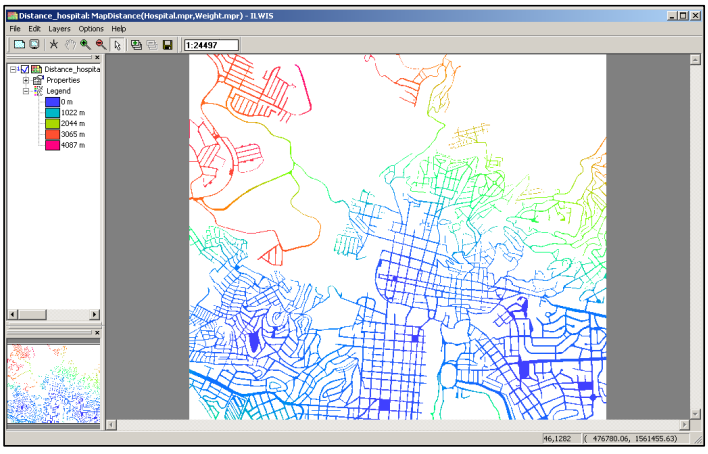
1. Creation of the attribute map of the hospital (that will be the input data on the distance evaluation)



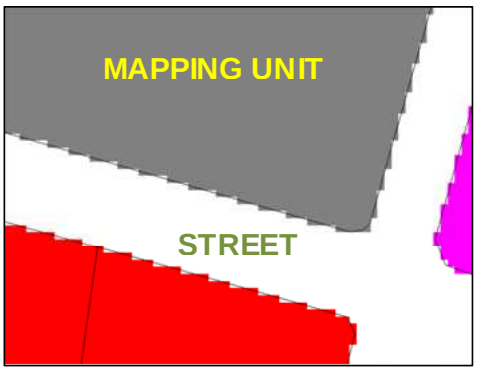


2. Creation of the weight map

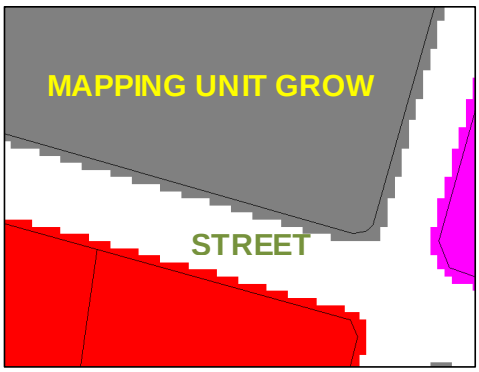
3. Generating a distance map (distance_hospitals).



We have only evaluated the distance along the street, pixel by pixel. We need to know the distance between every mapping units and the closest hospital. In order to do that we should cross the mapping units with the Distance_hospital map. This will give anyone results because the Distance hospital is evaluated along the streets, which are not intersecting the mapping units. For that reason we have to grow up at least of one pixel the dimension of the mapping units and make the mapping units crossable with the Distance_hospital. The images below show what has been just explained. Look at the border (in black) of the mapping_units.

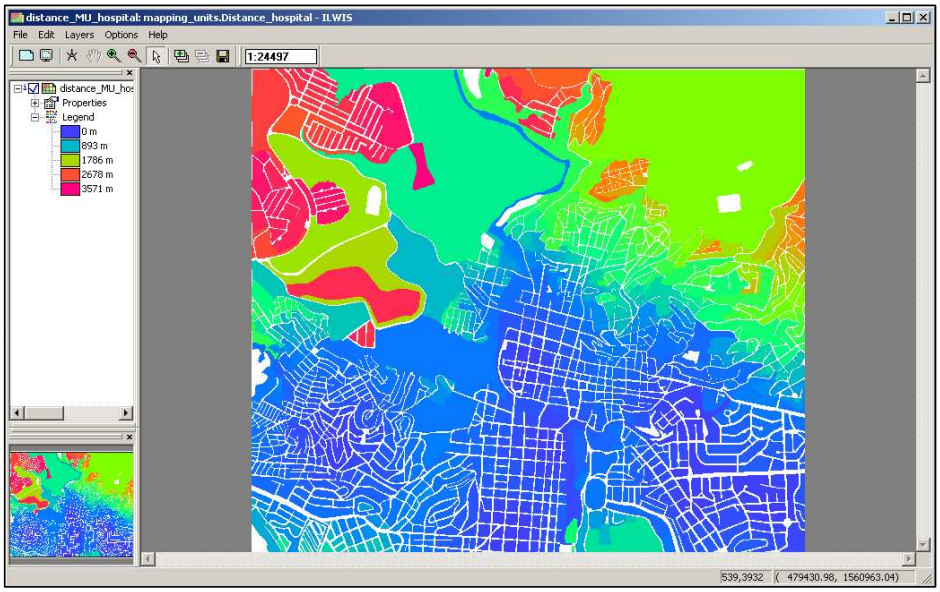


Mapping_units



Mapping_units_grow

The distance from every mapping units to the hospital is shown below. As you can see some mapping_units are undefined. This happens to the mapping units not surrounded by roads. (It is possible assign to this mapping units the value of distance to the hospital, using again the majority filter, but this time using the distance_MU_hospital.



The mapping units furthest away from the hospitals are mostly situated on the North-West of the area.

ADDITIONAL POSSIBILITY: taking in account the different degree of possibility to travel across the mapping units.

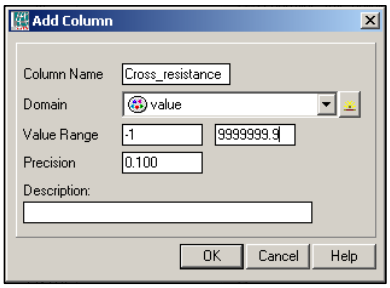
Table "landuse" - ILWIS

File Edit Columns Records View Help

	Builtup	Cross_resistance
Com_business	True	-1.0
Com_hotel	True	-1.0
Com_market	True	-0.8
Com_shop	True	-1.0
Ind_hazardous	True	-1.0
Ind_industries	True	-1.0
Ind_warehouse	True	-1.0
Ins_fire	True	-1.0
Ins_hospital	True	-1.0
Ins_office	True	-1.0
Ins_police	True	-1.0
Ins_school	True	-1.0
Pub_cemetery	False	0.5
Pub_cultural	True	-1.0
Pub_electricity	True	-1.0
Pub_religious	True	-1.0
Rec_flat_area	False	0.8
Rec_park	False	0.8
Rec_stadium	True	-1.0
Res_large	True	-1.0
Res_mod_single	True	-1.0
Res_multi	True	-1.0
Res_small_single	True	-1.0
Res_squatter	True	-0.9
River	False	-1.0
unknown	False	1.0
Vac_car	True	0.9
Vac_constructio	True	0.3
vac_damaged	True	0.2
Vac_shrubs	False	0.9

☞

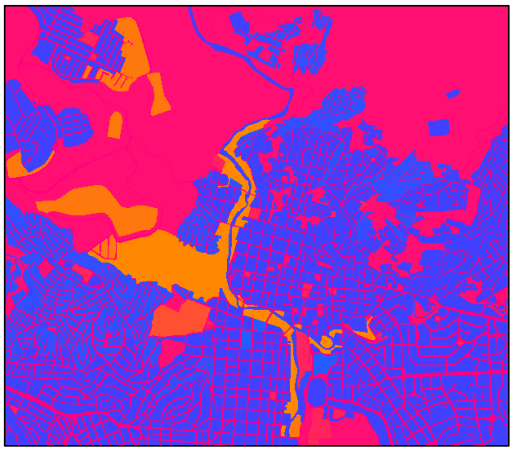
- Open the table Landuse, create a new column called **Cross_resistance**, use the value domain, a range between -1 and 1, and a precision of 0.1. Give a value to every landuse type, taking in mind that 0 means unpassable and 1 completely free. Use also the intermediate values. You can use the values shown on the left or assign your own values.



☞

- Open the table mapping_units and join with the column Cross_resistance from the table landuse. Call the output column with the same name.
- Create an attribute map of the column Cross_resistance from the map mapping_units and call it Cross_resistance.
- We need know to assign to the undefined values (in this case are the street) the value 1 (completely possible to travel).
- In the command line of ILWIS type the formula:

Weight_cross:=iff(isundef(Cross_resistance),1,Cross_resistance)



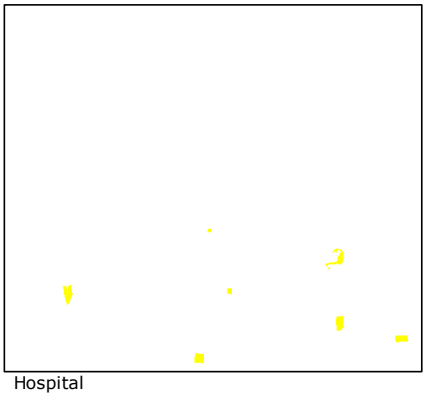
The map Weight_cross is shown on the left. The red area indicate area freely travelable while the blue are area not travelable. The orange are areas least travelable.

We will use now this map as weight for the distance calculation from the hospitals.

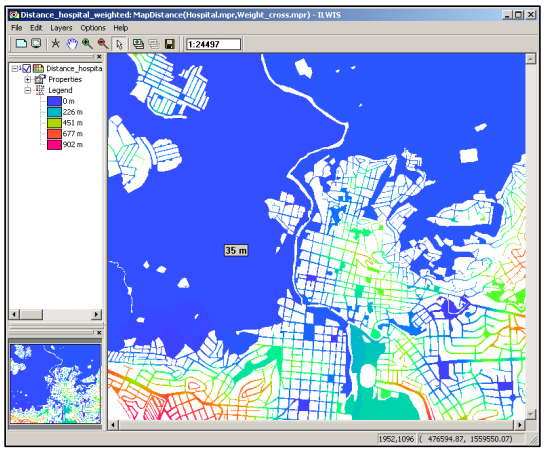
☞

- Go the distance operation in the operation list of ILWIS. Select the **Hospital** map as source map, the **Weight_cross** as weight and call the output **Distance_hospital_weighted**.
- Show the **Distance_hospital_weighted** map and check with the pixel information. Do you think is improved the calculation?

As you can see on the Distance_hospital_weighted there are some mapping units extremely big (Vac shrubs for example) that are considered as unit in the mapcalculation. This means that this areas have a values not realistic. In the exemple below there is the mapping units a value of 35 meters from the hospital. In other word, this value will be the same for the mapping units itself and influence even the calculation of the other areas (in fact areas close to this mapping units show values around 30-40.



Hospital



Distance_hospital_weighted

Anyway, we want to assign this values to the mapping units and later compare the results.

We need to do the same procedure made on the previous paragraph (cross the mapping_units_grow with the Distance_hospital_weighted).

☞

- Go to *operations, raster operations, cross* and select the **mapping_units_grow**, and the **Distance_hospital_weighted**. Call the output table **Mapping_units_G_W**.
- Now go to the **mapping_units** table and join with **Distance_hospital_weighted**. Use the minimum function and call the output column **Distance_MU_H_weighted**.

Normally the consideration that in some case you can also travel across the mapping units should not increase the distance from the hospital, but at least decrease. We want also check how much is decreased for every mapping units the distance.

☞

- Open the table **mapping_units** and type the formula:

check_wheight:=iff(distance_hospital_weighted>distance_hospital,1,0)

and check on the output column if there are "1" values (should not be).

- In order to figure out the meters gained crossing the mapping units, type the formula:

Meters_gained:=Distance_hospital-Distance_hospital_weighted

- Create an attribute map of **Meters_gained** and check the results.

As you can this is not a good approach, since is misleading to the reality. So we suggest to use the procedure applied on the previous paragraph.

Comment [j5]: If there is time at the end of the internship, try to develop a method that avoid this problem.

ADDITIONAL TIP: we are considering the distance from the fire station and police station. The procedure is the same used for the distance from the hospital evaluation.

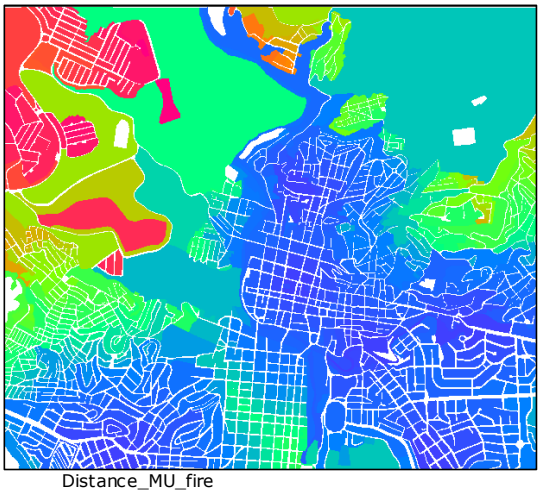
fire_station

☞

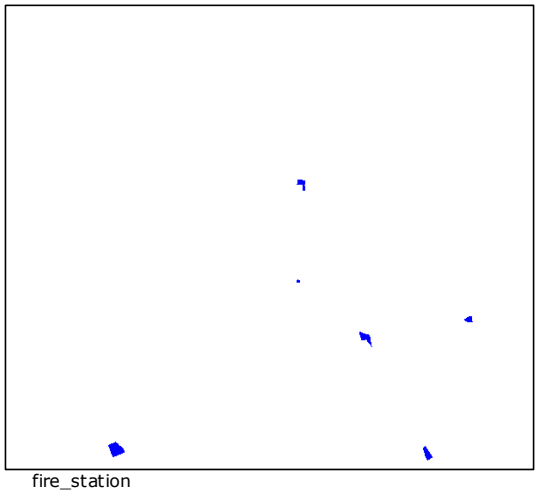
- Calculate in the table **mapping_units** a column **Fire_station**, with the following formula:

fire_station:=iff(Emergency_centers="fire_station",Emergency_centers,?)

- Make an attribute map **Fire_station** from this column and the raster map **Mapping_units.**
- We already have the Weight map from the previous exercise, so we do need to produce it again. We can directly measure the distance from the **fire_station**. Go to the distance operation, and select the **Fire_station** as source map and the **Weight** as a weight map. Call the output map **distance_firestation**.
- Cross the **mapping_unit_grow** with the **distance_firestation**. Call the output **map_grow_fire**.
- Open the table **mapping_units** and join with the table **map_grow_fire**. Read in the column **distance_hospital** and call the output **Distance_MU_fire**. Use the **minimum** function.
- Make an attribute map of **Distance_MU_fire** and call it with the same name.



Distance_MU_fire



fire_station

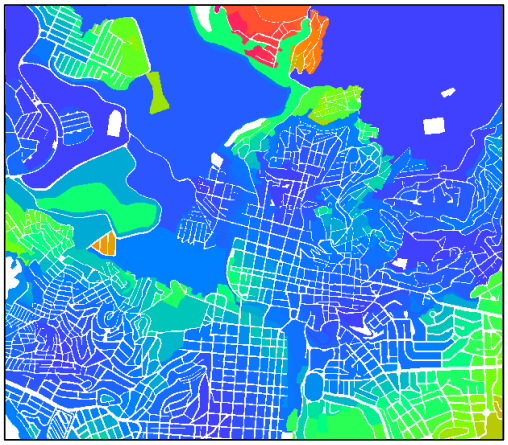
Police_station:

☞

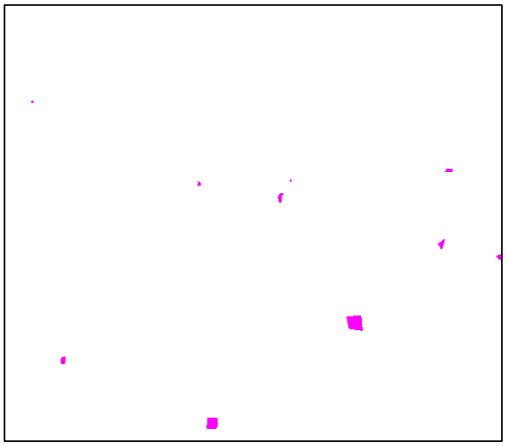
- Calculate in the table **mapping_units** a column **Police_station**, with the following formula:

Police_station:=iff(Emergency_centers="police_station",Emergency_centers, ?)

- Make an attribute map Pol**ice_station** from this column and the raster map **Mapping_units.**
- Go to the *distance operation*, and select the **Police _station** as source map and the **Weigh**t as a weight map. Call the output map **distance_ Police**.
- Cross the **mapping_unit_grow** with the **distance_ Police**. Call the output **map_grow_police**.
- Open the table **mapping_units** and join with the table **map_grow_police**. Read in the column **distance_police** and call the output **Distance_MU_police**. Use the **minimum** function.
- Make an attribute_map of **Distance_MU_police** and call it with the same name.
- Show the results.



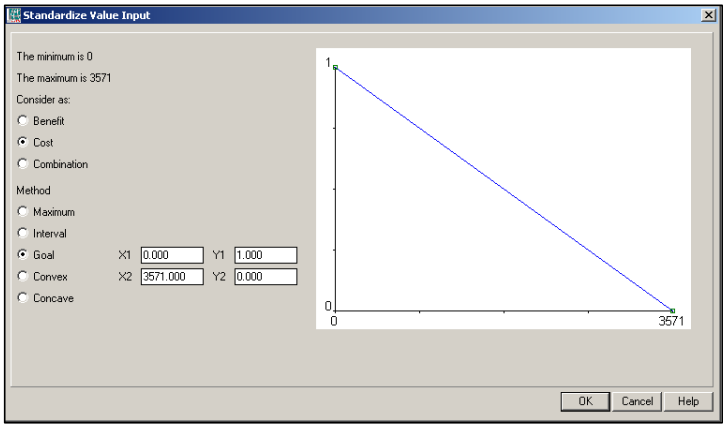
Distance_MU_police



Police_station

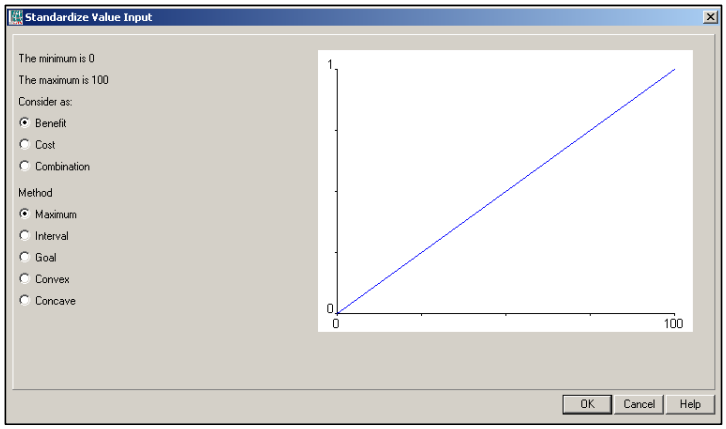
Generating the capacity indicator with SMCE

The image below shows the standardization of Distance to hospitals. In order to give a low value to the outmost mapping units, we used the cost function. As you can see with this options the function is opposite to the benefit option.

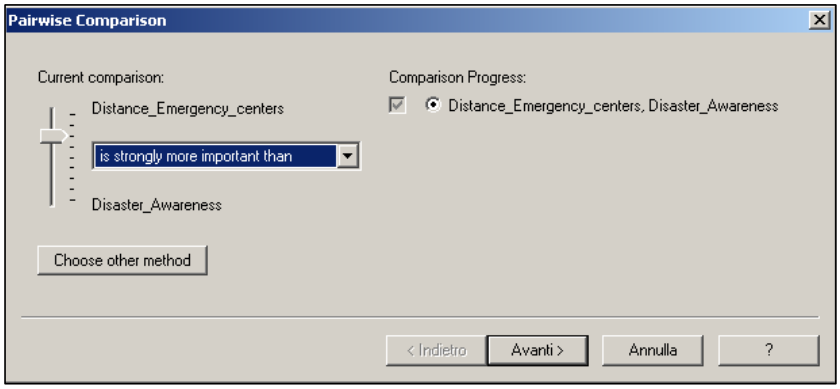


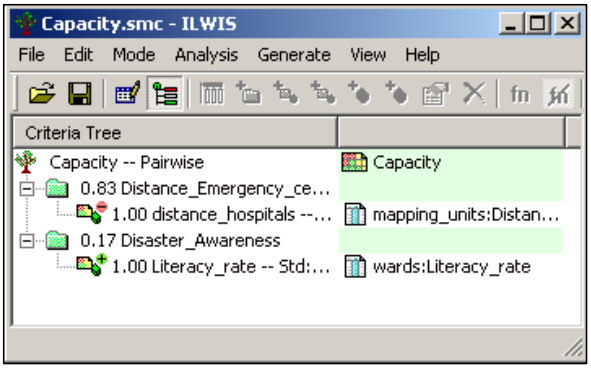
Note that the straight line of this image has a negative angular coefficient.

As you can see the standardization of the awareness has a trend opposite to the distance from the hospitals and used the Benefit function.

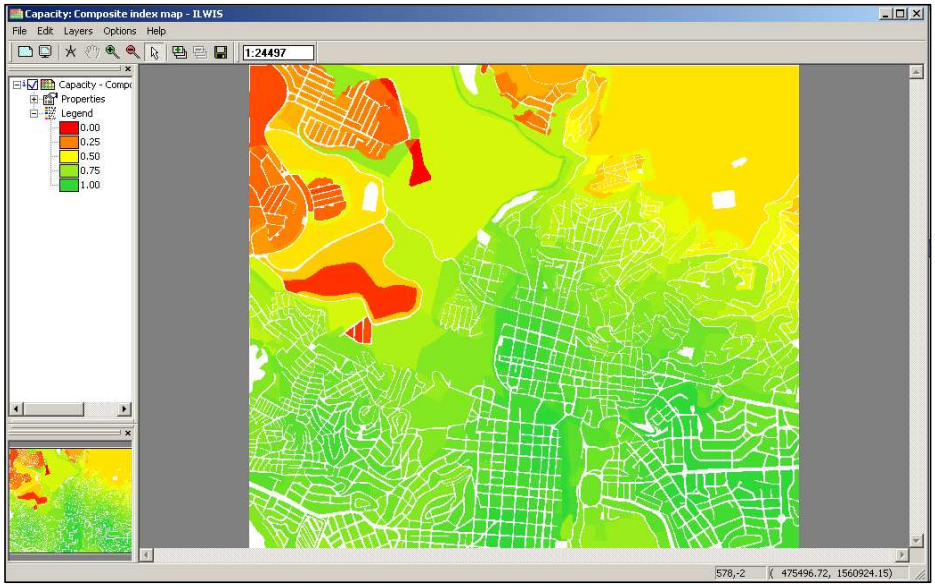


We considered the Distance from the hospitals more important than the awareness rate. Check the images below.





Combining the two capacity factor the Capacity map should looks like the following image.



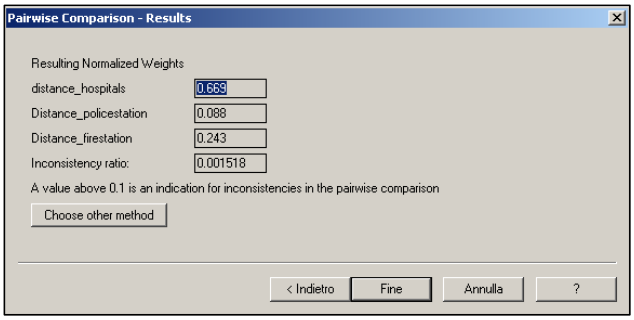
We considered here only the distance to the hospitals. The following exercise shows as include also the others distance maps.

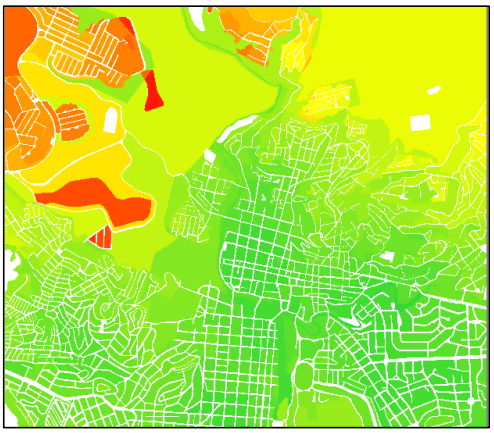
*** For experienced ILWIS users:**

Include in the capacity tree also the distance map from Fire_station and Police_station.

- Create another SMCE tree called Capacity_improved. Call the file name the same.
- Make the tree as the Capacity tree of the previous exercise.
- Add to the distance_emergency_centers two more factors: Distance_policestation, Distance_firestation.
- Standardize these factors and then the groups.
- Create the final map Capacity_improved, show the results and check with the Capacity map made on the previous exercise.

The pairwaise comparison between the distance to emergency centers is shown belows

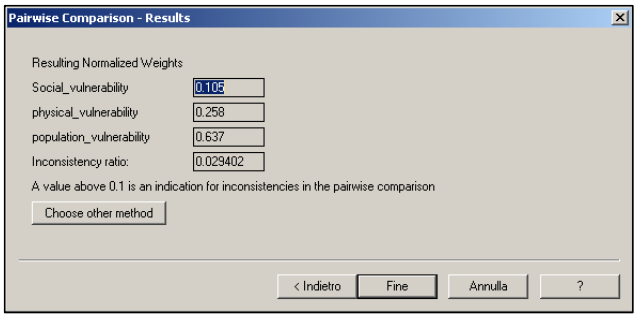


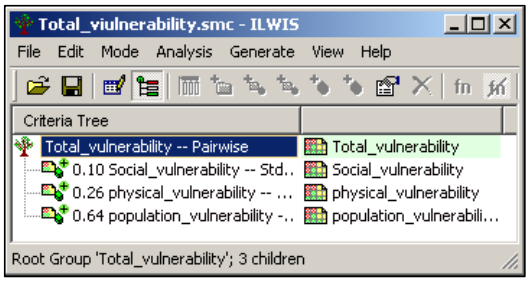


Capacity improved

5.6 Combing vulnerability and capacity indicators

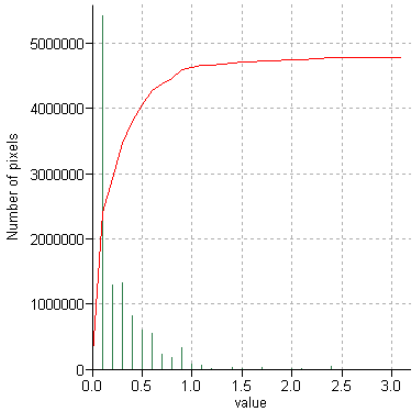
For the standardization among the tree vulnerability factor we used the maximum function. Check the image below in order to know the values used for the weigh.





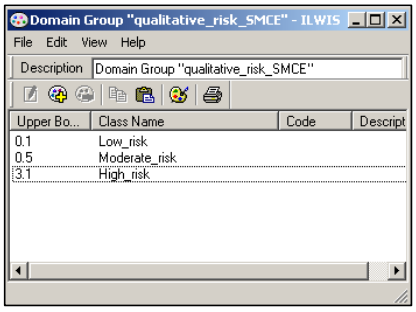
Total_vulnerability tree

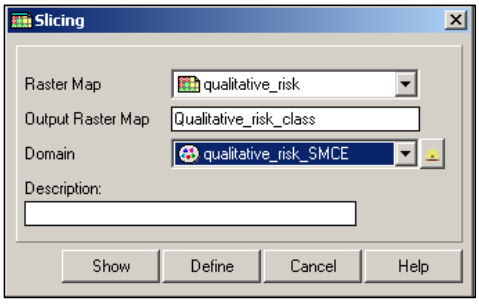
Before to reclassify the image on risk classes is useful have a look the Histogram of the image that have to be reclassified.



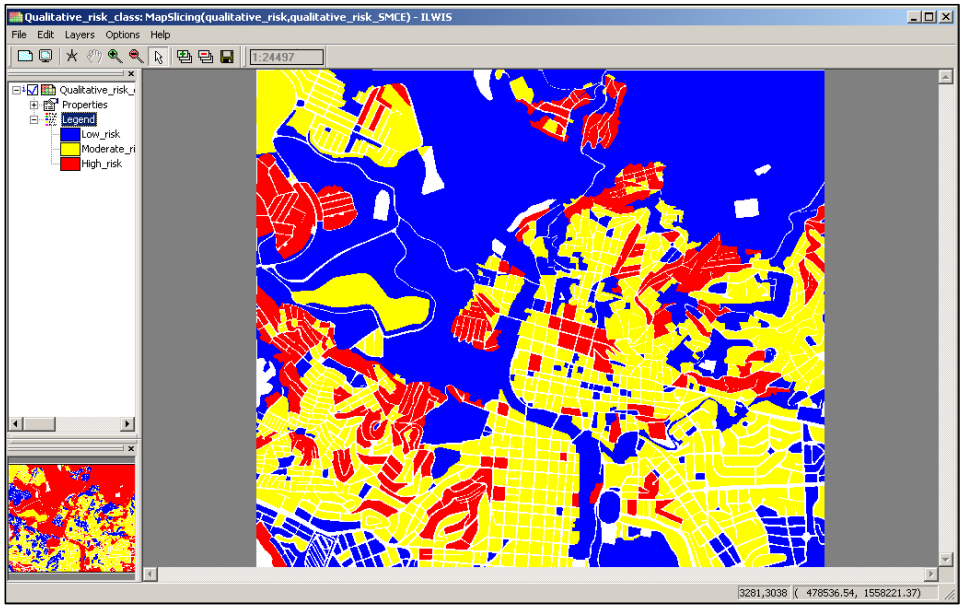
Histogram of qualitative_risk

It is possible to see the classes chosen on the image below of the domain qualitative_risk_SMCE.





The image below shows the qualitative risk map reclassified. Use the red color for the high risk, the yellow for the moderate risk and the blue for the low risk.



An alternative approach in using SMCE for vulnerability assessment

In this exercise we are going to redistribute the indicators available in ward and district on the mapping units. The assumption is that the percentage of the indicators in every mapping units is the same of the word of district that include it.

Social vulnerability.

Age_related

☞ **For experienced ILWIS users:**

Improve vulnerability map in the SMCE founding the evaluation on the mapping units

- Cross the mapping_units with the **District_map**. Call the output **mapping_units_district**.
- Open the table **Mapping_units** and join with the table **mapping_units_district**. Read in the column **Districts**. Call the output column **Districts**.
- In the table **mapping_units** aggregate the number of people and group per **districts**. Use the **sum** function and call the output **people_per_district.**
- Join the table **mapping_units** with the table **districts** and read in the columns: **Age_under_4, Age_4_to_12, Age_12_18, Age_18_24, Age_24_65, Age_over_65**. Use the same name for the output columns.

Comment [j6]: The idea is to redistribute the social vulnerability based on the nighttime population per mapping units

Now we can easily evaluate the factors per mapping units.

- Open the table **Mapping_units** and type the following formulas:

Age_under_4_MU:=Age_under_4*Nighttime_population/100
Age_4_to_12_MU:=Age_4_to_12*Nighttime_population/100
Age_12_18_MU:=Age_12_18*Nighttime_population/100
Age_18_24_MU:=Age_18_24*Nighttime_population/100
Age_24_65_MU:=Age_24_65*Nighttime_population/100
Age_over_65_MU:=Age_over_65*Nighttime_population/100

Later, include in the SMCE and in particular in the social vulnerability – Age related, all the factors created above.

Income related.

- Cross the **mapping_units** with the **Wards** map. Call the output **mapping_units_wards**.
- Open the table **Mapping_un**its and join with the table **mapping_units_wards**. Read in the column **wards**. Call the output column **wards.**
- Join also with the table **wards**, and read in the column **Unemployment**. Call the output with the same name.
- Again, join with the **column nighttime_population** from the table **wards**. Call the output column **nighttime_population_per_ward**.
- In the table **mapping_units** type the following formula:

Unemployment_MU:=Unemployment*nighttime_population_per_ward/100

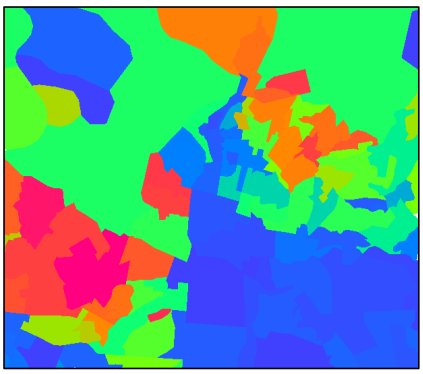
- Create an attribute map of **Literacy_rate_MU** and check the results.

Income related.

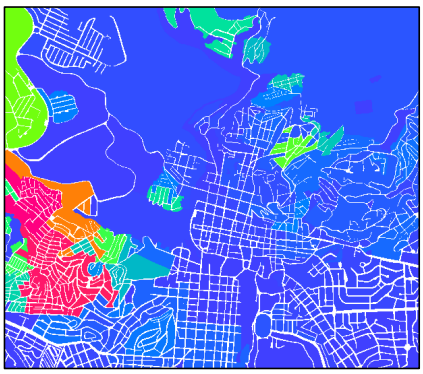
- Open the table **Mapping_units** and join with the table **wards**. Read in the column **Minority_groups**. Call the output column **Minority_groups.**
- In the table **mapping_units** type the following formula:

Minority_groups_MU:= Minority_groups *nighttime_population_per_ward/100

- Create an attribute map of **Minority_groups_MU** and check the results.



Unemployment



Unemployment_MU

Population Vulnerability.

For this factor the indicators are already at mapping units level.

Physical Vulnerability.

For this factor the indicators are already at mapping units level.

Capacity.

Disaster_awareness.

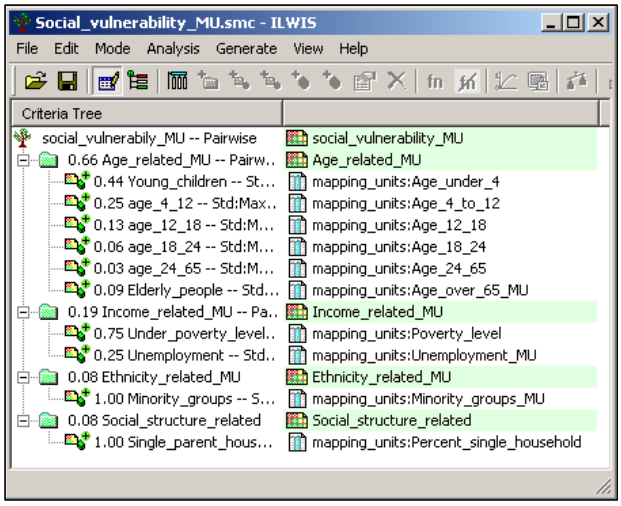
- Open the table **Mapping_units** and join with the table **wards**. Read in the column **literacy_rate**. Call the output column **literacy_rate.**
- In the table **mapping_units** type the following formula:

Literacy_rate_MU:=Literacy_rate*nighttime_population_per_ward/100

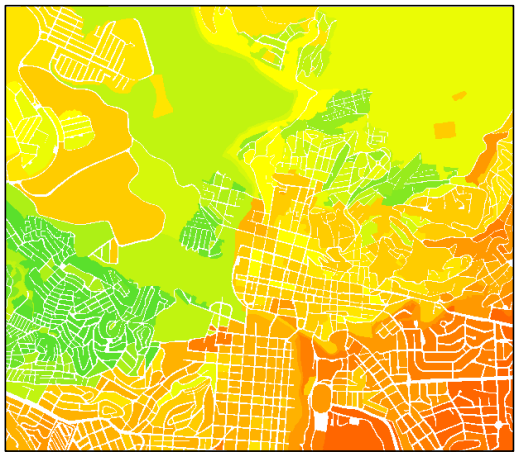
- Create an attribute map of **Literacy_rate_MU** and check the results.

Now you can create the vulnerability tree as made on the previous exercises using for all the factors, information at level of mapping units. The procedure is the same shown for all the exercise before.

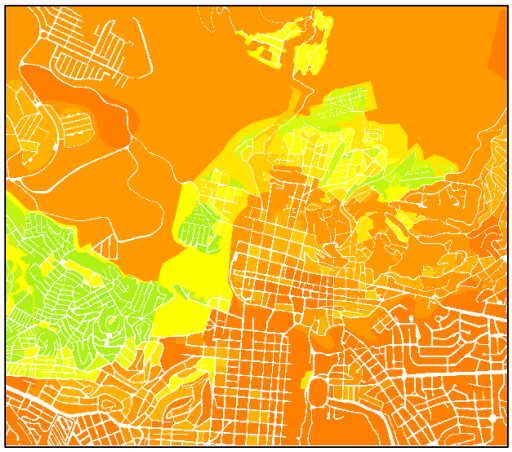
The social_vulnerability_MU tree should look like the following image.



and the social_vulnerabity_MU map:

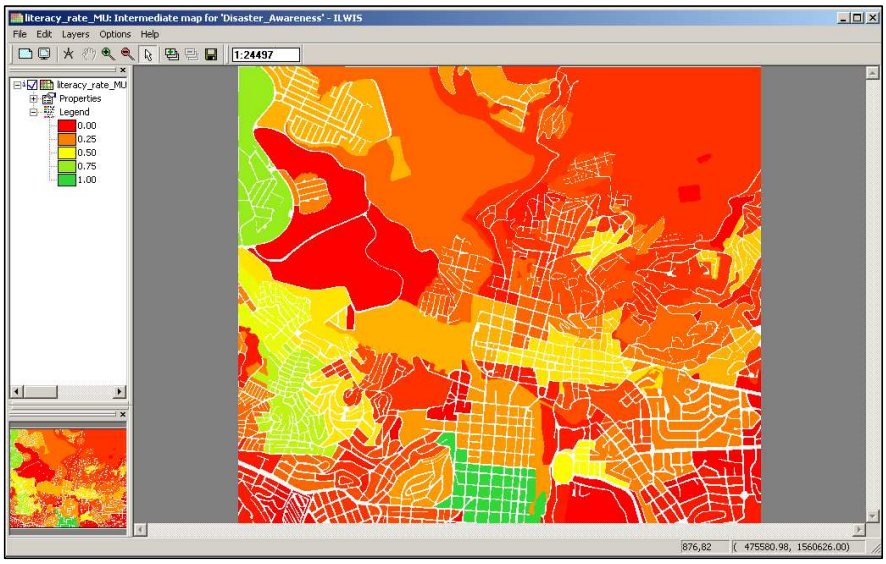


Social_vulnerability_MU

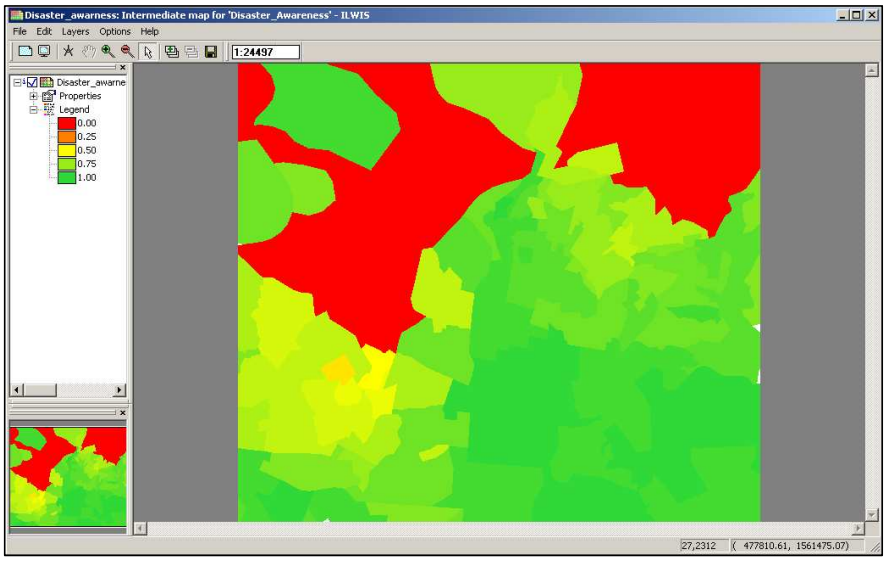


Social_vulnerability

It is easy to figure out that the resolution of the information is improved. On the Social_vulnerability, the shape of the classes follows the shape of the wards, and the values are almost constant on them, while on the Social_vulnerability_MU the information is more detailed and more variable.



Awareness_MU

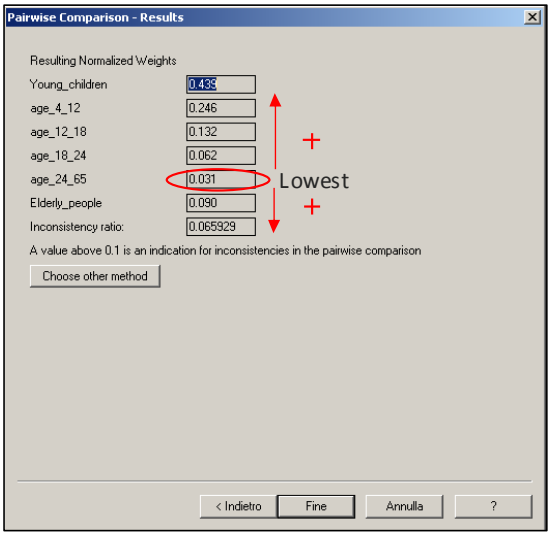


Disater_awareness.

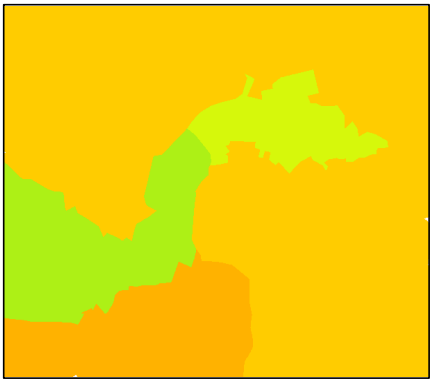
As you can see the result are different and much more detailed in the Awareness per Mapping Units.

On the age related group (from the social vulnerability) we are going to consider all the indicators create above. Include the following spatial factor: **Age_under_4, Age_4_to_12, Age_12_18, Age_18_24, Age_24_65, Age_over_65.**

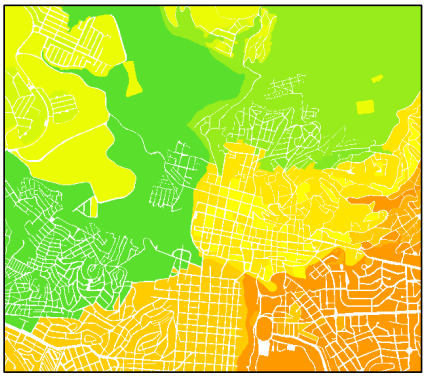
Standardize and weigh them as you retain opportune.



The images below show the differences between the Age_related (based on the wards) and Age_related per mapping units.

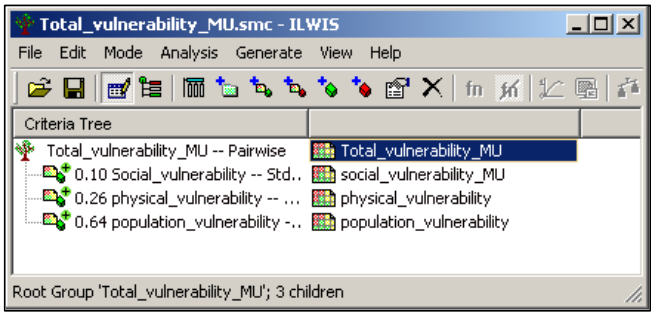


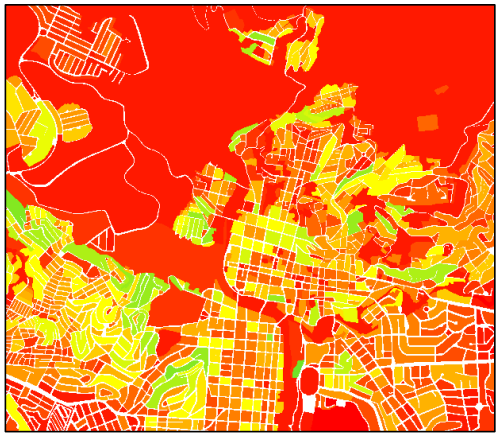
Age_related



Age_related_MU

Now it is possible combine everything and create a Total_vulnerability_MU tree.





Total_vulnerability_MU

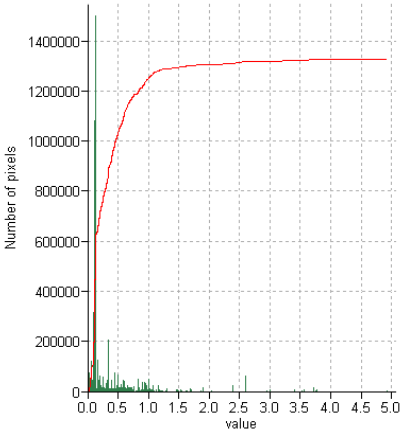
The total_vulnerability_MU doesn't change to much from the total_vulnerability of the previous exercise. This in part is done to the low weight assigned to the factor Social_vulnerability.

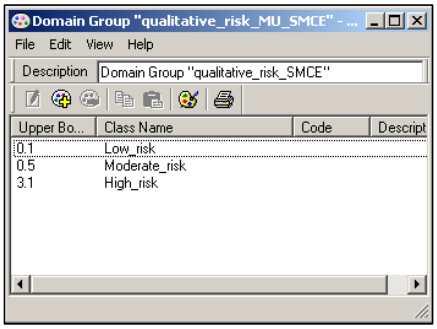
The last step is to evaluate the qualitative based on the information per mapping units.

- Type the formula:

Qualitative_risk_MU:=Total_vulnerability_MU/Capacity_Mu

  Use the value domain and a precision of 0.001
- Show the results.
- Create a new domain coping the **qualitative_risk_SMCE**, but modifying the upper boundary of the lowest class (or create a new domain with new classes according with histogram of **the qualitative_risk_MU** map). Call it **qualitative_risk_MU_SMCE**.
- Reclassify using the domain **qualitative_risk_MU_SMCE**. Call the new map **Qualitative_risk_MU_class**.
- Compare the results with the **Qualitative_risk_MU_class** creating the histogram of them.





Domain qualitative_risk_MU_SMCE Histogram of qualitative_risk_MU

As you can see from the histograms below, the final results change, and in particular with an increasing of the moderate risk class to detriment of the low risk class.