Interpolation by Inverse Distance Weighting

What’s that now, right? Well, i suppse in order for you and me to find out what that interpolation stuff is, first we’ll use a very well known tool(some call it Google) to define each of those pretty words in the title of the subject.

Soo, firstly, let’s find out what interpolation means. The word „interpolation”, according to dexOnline reffers to the intercalation and determination of some intermediary aproximative values of a function in a certain point of an interval, knowing the values of that same function in the extremities of the interval. –Huh? –I know, right? Shortly, the interpolation helps the estimation of some unknown values using the ones we already know(e.g.:let’s say we have two points in a 2-D space with known coordinates, and there comes a third point looking for it’s place in that Universe(as we all kinda do); we use those two points, draw a few lines and by some mathematical black magics, the point now found it’s place – the details will be discussed later on).

Now that we found out what the first term means, let’s talk about the reminder of that title.

Inverse Distance Weighting(or IDW as it’s friends call it), is one of the interpolation techniques. It explicitly implements the assumption that things that are close to eachother are more alike that those that are farther apart. To predict a value for any unmeasured location, IDW will use the measured values surrounding the prediction’s location.

IDW assumes that each measured point has a local influence that diminishes with distance. It weights the points closer t othe prediction location greather than those away, hence the name „inverse distance weighted”. Weight of each sample point, is an inverse proportion to the distance.

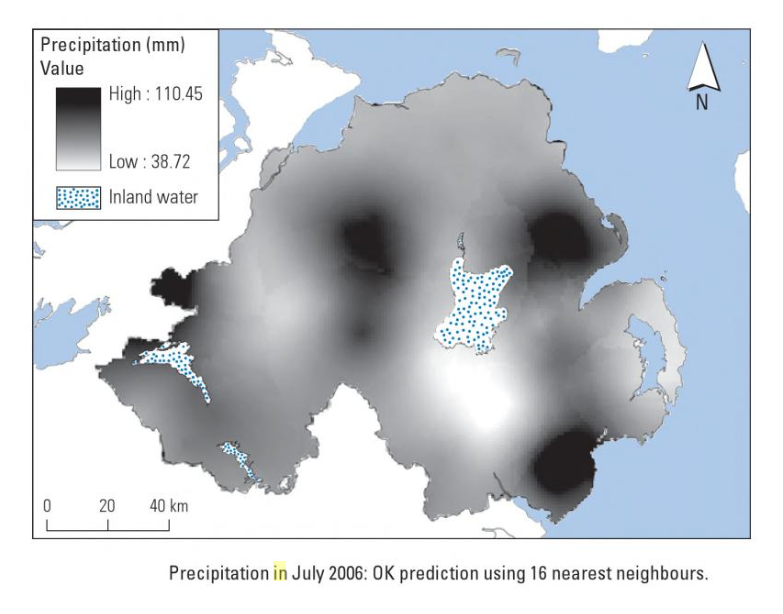
To give some kind of examples for the better understanding of the IDW:

* Noise is louder closer to a siren than further away.
* When it’s raining, it’s more likely to rain 1 meter away compared to 500 meters away.
* If you look 1 meter ahead, the terrain elevation is very likely to be the same. When you look 5 meters ahead, the ground elevation is going to be a little less similar.
* You got it…

These are examples of [**spatial autocorrelation**](https://gisgeography.com/spatial-autocorrelation-moran-i-gis/) or [**Tobler’s First Law of Geography**](https://gisgeography.com/tobler-first-law-of-geography/). Spatial autocorrelation is the underlying assumption of Inverse Distance Weighting.

The IDW isn’t just a mathematical thingy that someone invented out of boredom. It has some real life uses.

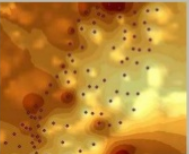
One clear example is the metheorological map that shows the quantity of rain over a certain piece of land. Of course, there isn’t someone in every point of the map to send information to a center in order to find if it’s raining in a place or not. There are a few metheorological stations sending information regarding the weather and using IDW and not only, a map that shows the predicted amount of rain is generated. Down below is a map of precipitation from 2006 generated using an algorithm based on the IDW with 16 nearest neighbours.



Another usage cases for the IDW could be:

* Methods for spatial prediction to estimate values at unsampled point locations which could be of interest.
* Time and money are imited; safety and accessibility restrictions.
* Small subsets of object, points or raster cells for estimating the total population.
* Loss of parts of collected samples (recovery) or unsuitability, obvious outliner points or just for closing „gaps.
* Prediction of variables at unmeasured locations based on a sample at unknown locations.
* Creation of surfaces of continuous values.

Examples:Temperature, productivity, elevation, population density, maps for video games, etc.



I know...for a thing you’ve never heard of, it actually can be used for many things.

Advantages for using IDW:

* It is clearly pretty easy to understand(only the title is kinda scary).
* Will not produce estimated outside measurements.

Disadvantages:

* No assessment to prediction errors.
* Produces „bulls eye” around data locations.
* The best results from IDW are obtained when sampling sufficiently dense with regard to the local variation you are attempting to simulate. If the sampling of input points is sparse or uneven the results may not sufficiently represent the desired surface.

Now that we’ve passed through all of these things, let us see how that method really works.

In order to use IDW, we first have to decide how do we pick the neighbours of the point that is about to be located. We therefore have two cases: either go with a **fixed search radius** or we use a **variable search radius**.

For the Fixed Search Radius, a neighourhood distance has to be specified. The distance of the radius is constant, so all the points that fall within the radius will be used in the calculation of the predicted point.

You can see in the Figure 1 the radius and the points inside it.

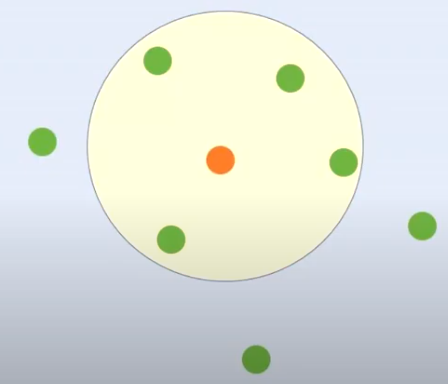
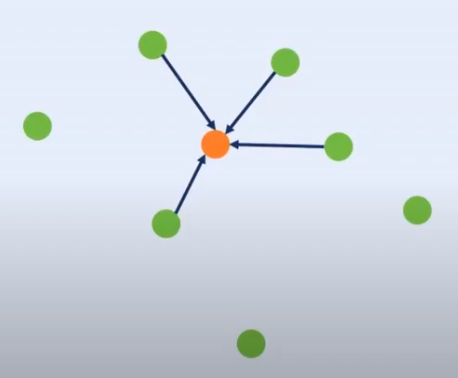
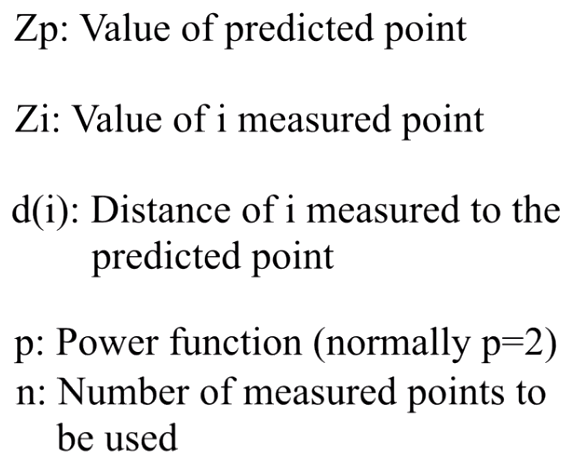
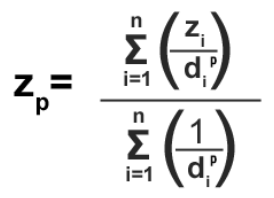
 

Figure 1 Figure 2

For the Variable Search Radius, what we need to specify is the number of measured points used in calculation. As an example, in the Figure 2 we tell IDW to use exactly 4 points in caluclations, it will choose the 4 points closest to the predicted point to use.

After setting that up, let’s see the abominable formula used to calculate the predicted point’s position. 

I know it doesn’t look pretty, deal with it!

As an example of how that formula looks like in practice:

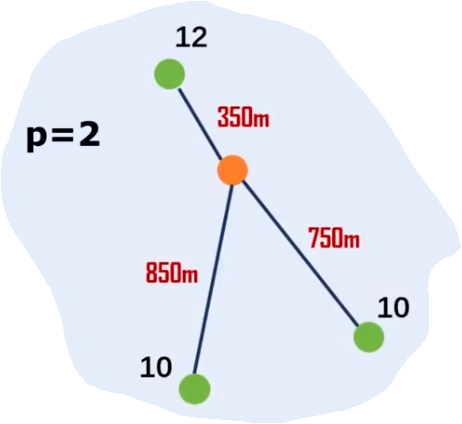
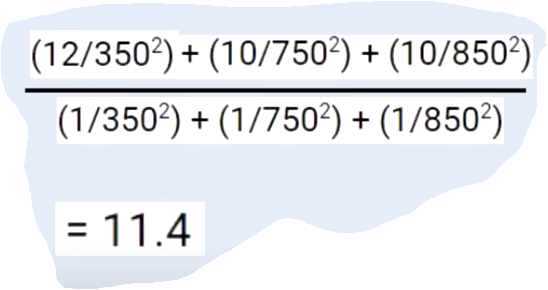
 

Figure 3 The cloud figure

So, there we have it - the mighty example! The red numbers represent the distances between the predicted point(orange) and the already known points(green), the numbers nearside the green points, represent the elevation of each one(like…something that works as a weight I suppose). In the cloud figure you can see the formula applied (the elevation divided by it’s distance to the power of two – but for each mesured point, all summed up and dividend to the same thing but all the elevations are replaced by 1 instead of their actual value. Finally, after the last division, we find the result – 11.4. A good way to find out if you suck at math is after computing the elevation of the point, verifiyng if it is in between the maimum and the minimum values of all the measured points. If it isn’t in that interval…you kinda need to do it all over again. What a pain, huh?

Thank you for reading, hope it helps!

Name: Vasiliu Gheorghe-Adelin

Group: E2

Bibliography:

[Youtube video – Inverse Distance Interpolation Method](https://www.youtube.com/watch?v=3eoEWcxXBNs)

[Inverse Distance Weighting (IDW) Interpolation](https://www.youtube.com/watch?v=3eoEWcxXBNs)

[Christopher D. Lloyd – Spatial data analysis – an introduction for GIS users](https://books.google.ro/books?id=7WGcAQAAQBAJ&pg=PA149&lpg=PA149&dq=IDW+examples+in+real+life+use&source=bl&ots=M-_izevTSu&sig=ACfU3U3TcDmq4ELPns42AVapgCtcf0Mxvg&hl=ro&sa=X&ved=2ahUKEwjZvMDR-LbpAhVK2aYKHbytCD0Q6AEwAHoECAoQAQ#v=onepage&q=IDW%20examples%20in%20real%20life%20use&f=false)