GIS-4 Interpolation

2025-04-13

Per Halvorsen
pmhalvor@uio.no
GEO4460

1 Introduction

In this report, we will analyze the relationship between snow depth and thaw depth in Adventsdalen, Svalbard. Specifically, we will:

- Build and compare interpolation layers of snow and thaw depth data
- Analyze how the data changes over time
- Construct linear regression models to predict thaw depth in N months based on snow depth

The goal of this report is to build an understanding of different interpolation methods, and how they can be used to analyze spatial data. The interpolation methods to be comapred are global polynomial interpolation, inverse distance weighting, and kriging.

2 Data

The data used for this analysis is the monthly thaw and snow measurements from Svalbard over the summer of 2007. These data were collected by the Norwegian Polar Institute and are available in the `data` folder.

The data contains measurements over an $11x11$ grid of evenly spaced points in the Adventsdalen valley, though not directly tied to any specific location longitude and latitude. This means the analysis will need to be run on without an explicit coordinate system, but rather on a grid of points.

The data is stored in a DBF file, which contains the following columns: - `XM`: The x-coordinate of the measurement location - `YM`: The y-coordinate of the measurement location - `SNOW200705`: Snow depth in May 2007 - `THAW200705`: Thaw depth in May 2007 - `THAW200706`: Thaw depth in June 2007 - `THAW200707`: Thaw depth in July 2007 - `THAW200708`: Thaw depth in August 2007 - `THAW200709`: Thaw depth in September 2007

3 Method

3.1 Interpolation

As mentioned above, we will be comparing three different interpolation methods: global polynomial interpolation, inverse distance weighting, and kriging. Each interpolation method can be

implemented in ArcGIS using the Geostatistical Wizard. When configuring the interpolation, each method comes with unique parameters that need to be optimally set in order to achieve the best results.

3.1.1 Global/Local Polynomial Interpolation (GPI/LPI)

Being the simplest of our interpolation models, global and local polynomials are fast and efficient approximations, making them ideal for situations with densely populated data that needs to be compressed or simplified. Global polynomials are more likely to focus on larger trends across the data, while local polynomials enable more granular representations. Splines are an example of local polynomials that are relatively robust against clustered data, as long as sampling can be considered fair — for example, if sparse measurements occur in smoothed terrain with little variance.

Global polynomials are usually coarse representations, making them typically very underfit compared to the other interpolation methods we've observed. Local polynomials are prone to overfitting and can also be quite sensitive to parameter tuning, similar to IDW. In areas of complex terrain, they may oscillate or produce artifacts that don't reflect realistic surface features.

3.1.2 Inverse Distance Weighting (IDW)

Inverse distance weighting can be a simple and exact approach to elevation mapping, requiring only a min/max neighborhood size and a power coefficient. Synonymous with a circus tent, with the internal poles representing the measured input elevations, these models are often highly intuitive. Cells of an inverse distance weighted raster are calculated from, and influenced by, their closeness to their neighbors. This preserves the local variation captured by the measurements and is typically considered robust, even over sparsely populated regions in an otherwise very clustered dataset.

A flaw of highly localized models is that they tend to focus only on a point's immediate neighborhood and thus fail to pick up on any global trends, making them less adaptable to complex terrain. This can lead to heavily biased representations if sampling is not fairly distributed. Proper tuning is important when using IDW, since the power parameter and neighborhood sizes can be very sensitive to overfitting. To ensure generalizability, it's good practice to tune IDW models using a validation set, ensuring the optimal parameters also achieve reasonable errors on unseen data.

3.1.3 Kriging

Using the semivariance between points, kriging finds the optimal relation between how much neighboring points in the input data should influence each other. This not only makes kriging the most statistically optimal interpolation method, but also provides a variance surface, which represents each cell's corresponding certainty. Localized predictions with certainty estimations make kriging a great method for use-cases that require fine-grained resolution with limited or sparse data.

Due to being an abstract statistic on the data, interpretability of kriging can be more cumbersome than other simpler models. To understand the trends the model is picking up on, an analyst needs to be able to interpret a semivariogram, which is not always directly intuitive. Kriging's added complexity of fitting a variogram model and solving a system of equations for each prediction makes calculating these interpolations slower than other methods, especially when working with large data sets. These equations assume the input data has some innate statistical predictability, which might not always be the case — for example, with very noisy data, highly irregular sampling patterns, or human-influenced landscapes.

3.2 Geoanalysis

In this section, we will analyze the trends we see in the snow melt data. This analysis can be done using the raw data from the DBF file or randomly selected points from our rasters, both producing very similar results. The major difference between the two analyses would the amount of outliers in the data raw data, which might introduce bias and skew the statistical analysis. Even though it requires a few extra steps, we'll be using sampled data from the rasters, to document this process as well, for future reference.

For this analysis, we will leverage open-sourced Python libraries such as `numpy`, `pandas`, and `matplotlib` for efficient and reproducible results.

3.2.1 Trends in thaw data

```
[1]: from utils import load_snow_melt_data_raw
     from utils import load_snow_melt_data_sampled

     snow_melt = load_snow_melt_data_sampled(file_name="sample_raster_snow_melt.csv")
     columns_of_interest = [col for col in snow_melt.columns if col.
       ↪startswith('SNOW') or col.startswith('THAW')]

     snow_melt[columns_of_interest].describe()
```

```
[1]:        SNOW200705  THAW200705  THAW200706  THAW200707  THAW200708  THAW200709
     count  100.000000   100.0000  100.000000  100.000000  100.000000  100.000000
     mean     8.137469    34.1700   68.221242   94.388435  103.472396  104.590000
     std      3.204180     9.2715    2.687098    4.755914    2.975540    6.167846
     min      1.005777     1.0000   65.011284   86.177933   97.351036   92.000000
     25%      6.765413    28.0000   66.244072   90.114353  101.165003  100.000000
     50%      8.055214    35.5000   67.369930   93.794960  104.141502  105.000000
     75%     10.690675    40.0000   69.706110   98.603336  105.640558  108.000000
     max     15.080025    72.0000   76.517662  101.919251  108.231163  121.000000
```

Table 1: Statistical summary of raster samples' snow melt data.

```
[2]: snow_melt_raw = load_snow_melt_data_raw(dir_path="GIS4_datafiles",␣
       ↪file_name="month_thaw_snow.dbf")
     snow_melt_raw[columns_of_interest].describe()
```

```
[2]:        SNOW200705  THAW200705  THAW200706  THAW200707  THAW200708  THAW200709
     count  121.000000  121.000000  121.000000  121.000000  121.000000  121.000000
     mean     8.099174   33.603306   69.694215   94.570248  103.545455  104.561983
     std      3.880303    9.242546    7.939189    6.726597    5.599107    6.154798
     min      0.000000    1.000000   55.000000   80.000000   89.000000   90.000000
     25%      6.000000   28.000000   65.000000   89.000000  100.000000  100.000000
     50%      8.000000   35.000000   68.000000   95.000000  104.000000  105.000000
     75%     11.000000   39.000000   72.000000  100.000000  107.000000  108.000000
     max     17.000000   72.000000  100.000000  110.000000  119.000000  121.000000
```

Table 2: Statistical summary of true samples' snow melt data.

It is important to double check the prediciton data against the true values. Since we've sampled the rasters in the same regions that we previously have measurements, the predicted values should be similar to the true values for each data set. Judging by the statistical summaries inte tables 1 and 2, showing prediction data and raw measure respectively, we can see that the predicted values are very similar to the true values. This is a good sign, as it indicates that the interpolation methods used is a good representation of these data.

```
[3]: from plot import boxplot_snow_melt

    boxplot_snow_melt(snow_melt)
```
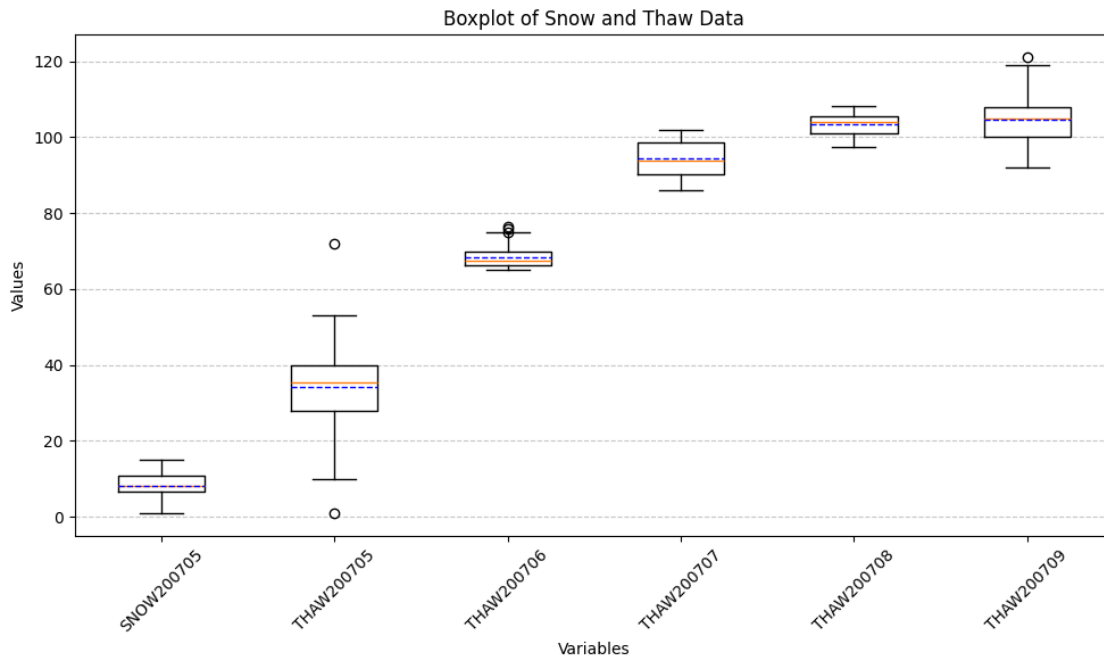

Figure 1: Box plots show the distribution of thaw data over the 5 months of measurements. The wider the spread, the greater the variation in thaw data across the 11×11 grid. The yellow line shows the median value at each month, while the blue dotted line shows the mean value. The lower and upper limits of the box show the 25th (Q_1) and 75th (Q_3) percentiles of the data, respectively. The whiskers show the range of $1.5 \times IQR$, where $IQR = Q_3 - Q_1$. The circles dots outside the whiskers are the outliers in the data.

```
[4]: from plot import plot_snow_melt_statistics

    plot_snow_melt_statistics(snow_melt)
```
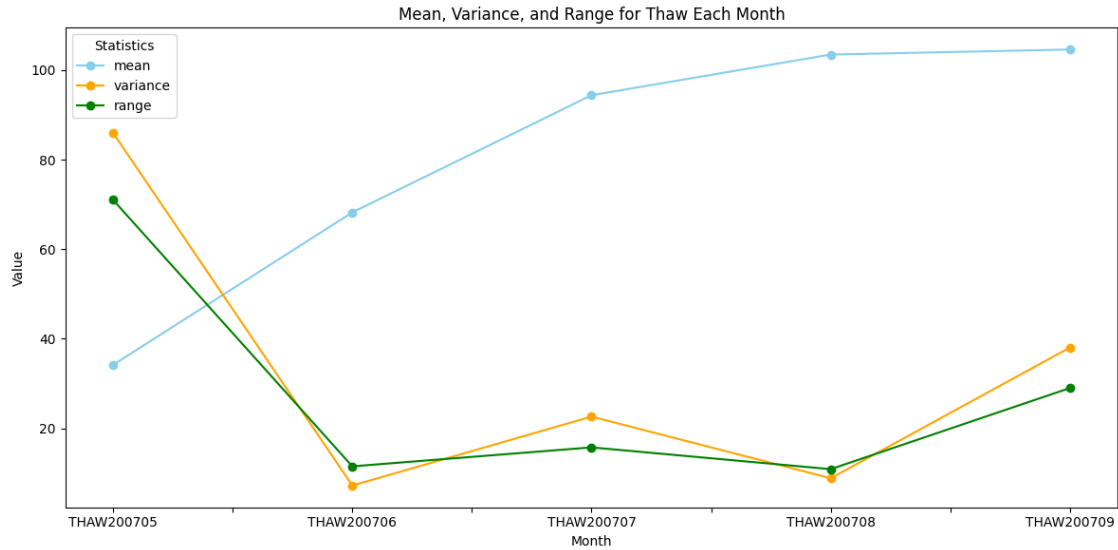
4

Figure 2: A simplified display of the description data from the box plot. Here, only the mean, variance and ranges for each month are shown. As the summer progresses, the mean thaw depth increases, while the variance and range of the measurements decrease. The flattening of all the curves around August and September tells us the system is becoming more stable as the summer progresses.

3.2.2 Difference in active thaw depth versus snow depth

```
[5]: from plot import plot_snow_melt_regression_lines

     regression_lines = plot_snow_melt_regression_lines(snow_melt)
```
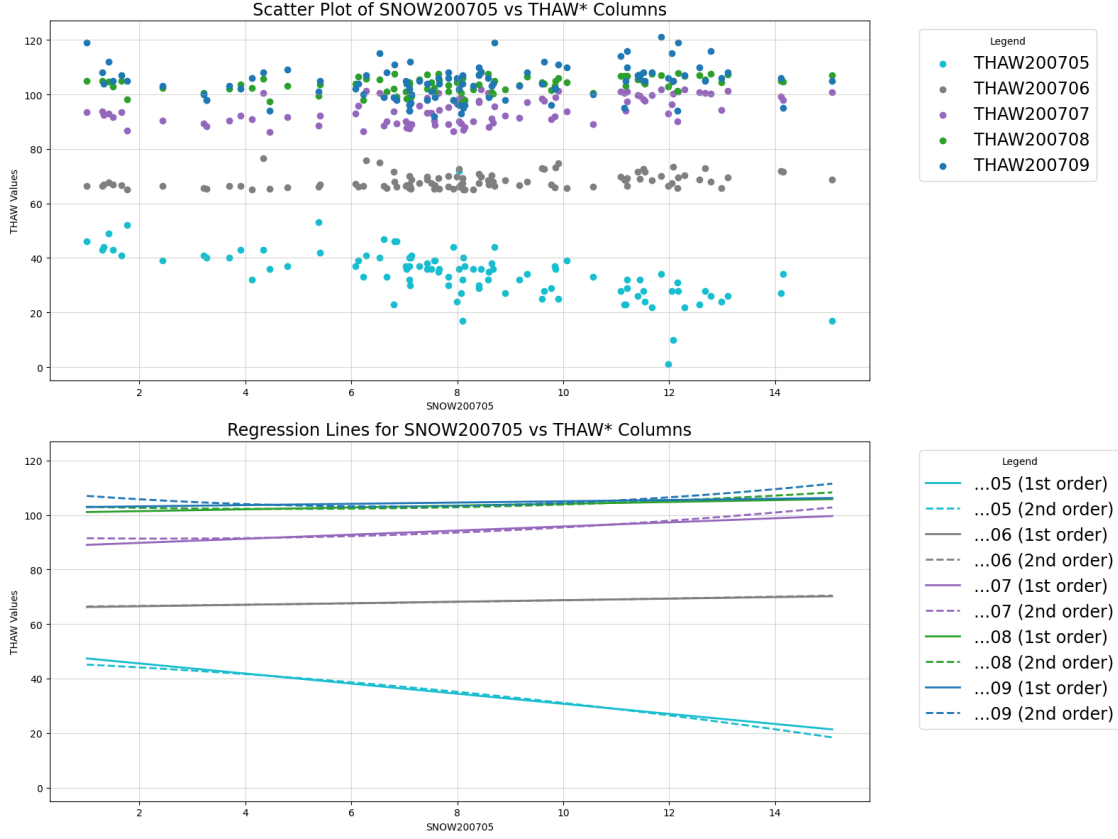
<Figure size 1200x800 with 0 Axes>

Figure 3: The two plots show the relationship between snow depth in May (x-axis) and thaw depth in the respective months (y-axis). The randomly sampled datapoints are shown in the scatter plot on top, and the 1st and 2nd order regression lines that can be used to predict the thaw depths' at each more is shown in the bottom plot. Initial analysis shows that higher inital snow depth typically gives lower thaw depth early on, and that thaw depths increase as the summer progresses, stabilizing around August and September. More in the analysis section.

```
[6]: from utils import process_regression_data

     process_regression_data(regression_lines)
```

```
[6]:     THAW Column          1st Order                    2nd Order
     0  THAW200705   [-1.8499, 49.2232]    [-0.0661, -0.8346, 46.007]
     1  THAW200706    [0.2803, 65.9406]     [0.0057, 0.1924, 66.219]
     2  THAW200707    [0.7517, 88.2712]    [0.0714, -0.3456, 91.7475]
     3  THAW200708   [0.3369, 100.7308]     [0.056, -0.5243, 103.459]
     4  THAW200709    [0.2341, 102.685]    [0.119, -1.5956, 108.4811]
```

Table 3: 1st & 2nd order polynimal parameters describing the regression lines plotted in Figure X, in format $y = ax^2 + bx + c$ for a second order polynomial in the table as [a, b, c].

4   Analysis

4.1   Interpolation

Data	Method	Root-mean squared
SNOW200705	*Kriging*	*3.2978*
SNOW200705	IDW	3.6789
SNOW200705	Global (1st)	3.8067
SNOW200705	Global (4th)	3.6635
THAW200705	*Kriging*	*7.6265*
THAW200705	IDW	7.9546
THAW200705	Global (4th)	8.2112
THAW200709	*Kriging*	*5.3182*
THAW200709	IDW	5.4127
THAW200709	Global (4th)	5.4672

Table 4: The root mean square (RMS) error of the different interpolation methods on snow and thaw data.The lower the RMS, the better the interpolation method. The same parameters were used to build each method for all three data sets.

The interpolator that minimizes RMS error is usually the best fit for the domain being analyzed*. RMS error is a measure of how well the interpolator fits the input data, and the lower the RMS, the better the fit. Table 4 show that across the board, kriging performed best, producing the lowest RMS error on all three datasets.

It's worth noting that an even better fit might be achieved through more extensive parameter tuning. In this case, the default kriging parameters were used, with only minor adjustments.

Both IDW and global polynomial interpolation (GPI) had noticeably higher RMS errors than kriging. This suggests the data contains both global trends and local variations that kriging is able to account for — but that IDW and GPI are not well-equipped to capture simultaneously.

To test the flexibility of GPI, we compared both 1st and 4th order polynomial fits on the snow dataset. We found that the 4th-order polynomial provided the best results across all datasets, so that version was selected for the main analysis. The 1st-order fit is included as a reference for how the model behaves under a simple linear assumption.
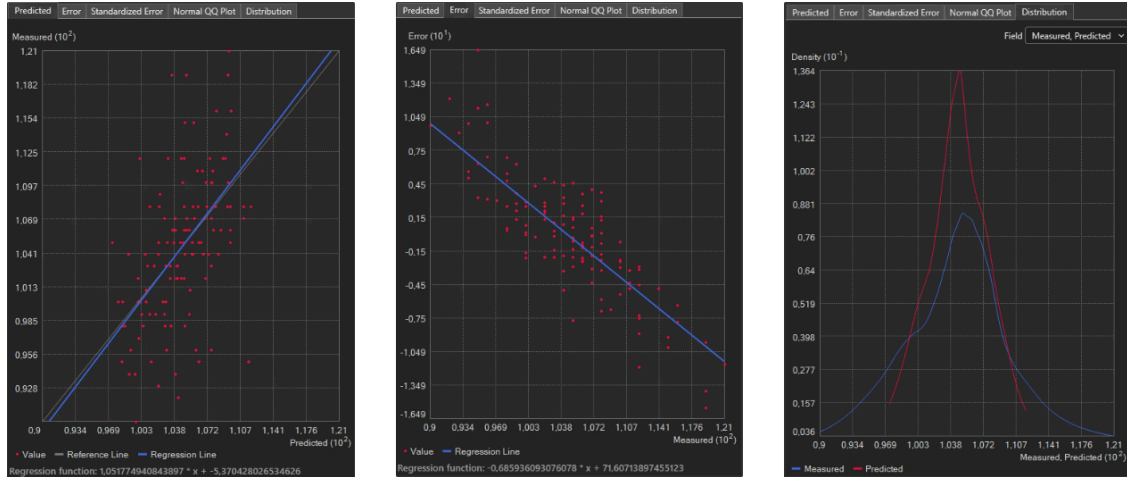
Figure 4: The three plots shown for most interpolation methods when fitting in ArcGIS, including prediction & error scatter plots w/ residual trends and a distribution comparison between predictions and measurements. These plots are from the kriging interpolation layer for the THAW200709 data.

When building any interpolation method, ArcGIS provides diagnostic plots of predictions, errors, and distribution comparisons to give users more insight into how well the model fits the data. The residual lines and scatter plots are similar to what we looked at in the Difference in active thaw depth versus snow depth section, with the key difference being that the points here known values, not the sampled raster predictions.

In an ideal case, the points would lie perfectly along the prediction line, indicating a flawless model fit. But real-world terrain is rarely that smooth or noise-free.

I found the distribution plots (shown to the right in Figure 4) to be especially informative. These show how well the predicted values match the spread of the actual measurements. In nearly every model I looked at, the predicted distribution (marked by a red line) had a taller, narrower peak around the mean and covered a smaller portion of the x-axis than the measured data. This indicates that all of our interpolation models tend to predict values closer to the mean, and struggle to accurately estimate extreme values at either end of the range.

A caveat to using RMS as the sole selection criterion for choosing the best interpolator is the risk of overfitting. A model that perfectly matches a given dataset may not generalize well to new or unseen measurements. Interpolators like IDW are especially prone to overfitting. This is often visually apparent when the interpolation surface shows perfect circular influence zones around each sample point.
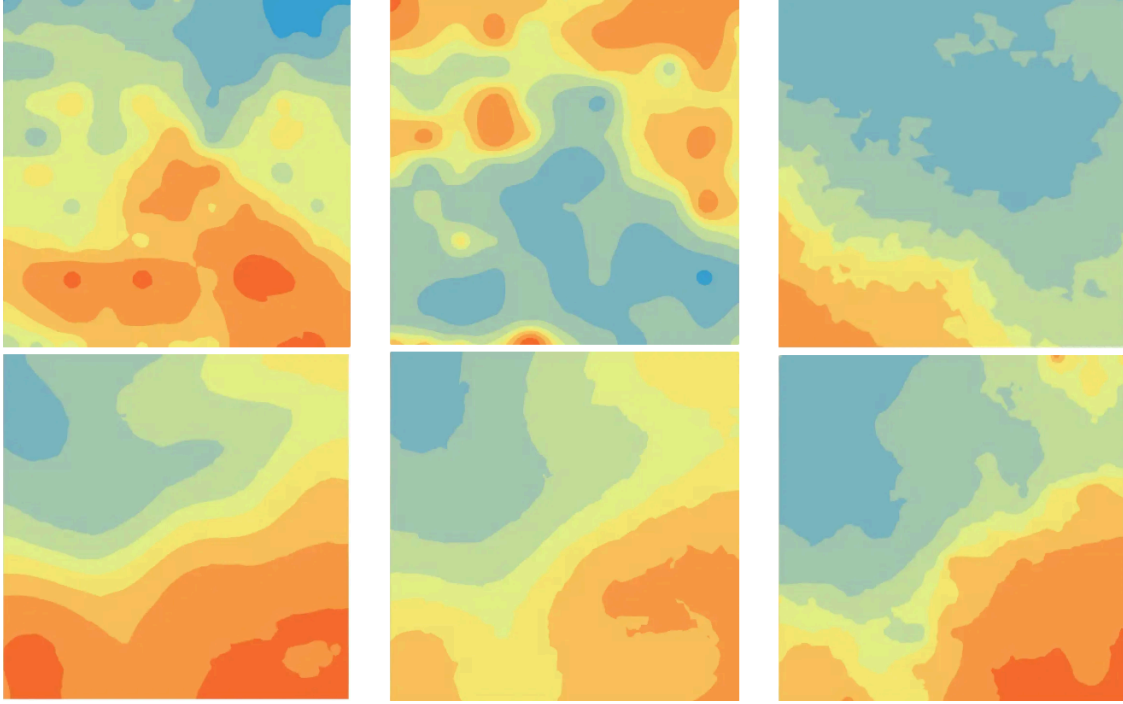
Figure 5: The 6 kriging interpolation layers generated for this analysis. From left to right, the top row shows the snow depth in May, thaw depth in May, and thaw depth in June. The bottom shows thaw depth in July, August, and September. The color scale varies for each layer, depending on that month's range. These can be found using the boxplot in Figure 1.

4.2 Geoanalysis

4.2.1 Analyzing trends in thaw data

Figures 1 and 2 show the trend of mean thaw depths over the summer months of 2007, along with some other useful statistics. We see that as the summer progresses, the mean thaw depth increases, eventually plateauing around August and September. The variance and range of thaw depths also decrease over time, indicating that the system becomes more stable as the summer progresses.

The boxplots show that there are not many outliers in the sampled data, with the exception of the first month, May. In the raw data, this month also showed high variation, suggesting that May is a volatile month for thaw depth — thawing is not yet uniformly distributed across the grid.

In September, the boxplots show some values lower than those from the previous month, indicating that some thawed ground may be starting to refreeze. This, combined with the plateauing of the mean thaw depth, suggests that September marks the end of the thaw season and the beginning of the next freeze cycle.

4.2.2 Analyzing the relationship between snow depth and thaw depth

Figure 3 shows the relationship between snow depth in May and thaw depth in the respective months. As seen in Figures 1 and 2, the thaw depth increases as the summer progresses, so the sample points and residual lines also move up along the y-axis for each consecutive month. We can also observe the same plateauing effect around August and September in the thaw depth as we

saw in the boxplots, with many overlapping points in this regions of those months' corresponding color-coded scatter plots.

The lowest thaw depth is again seen in May, but here with a clear downward trend in thaw depth as snow depth increases. This is also observed in table 3, where the 1st order regression line parameters shows a negative slope for May only, indicating that thaw depth decreases as snow depth increases. This means that areas with high snow depth at the beginning of the thaw season are likely to initially thaw less than areas with lower snow depth. This is likely due to the insulating effect of snow, which prevents the ground from thawing as quickly.

An interesting finding from the 2nd order regression analysis was the slight *increase* in thaw depth at the highest snow depths for the later months. However, given we have previously concluded that outliers are not necessarily well represented in the data, this finding could be an artifact of the predictions, rather than a real-world phenomenon. While this pattern also appears when building residual lines for the raw data, We would still need to consult with a geologist to confirm or refute this hypothesis.

5 Conclusion

In this report, we've looked into the relationship between snow depth and thaw depth in Adventsdalen, Svalbard. We built and compared three different interpolation methods: global polynomial interpolation, inverse distance weighting, and kriging. Kriging was found to be the best method for this dataset, with the lowest RMS error across all three datasets.

We also analyzed the trends in thaw depth over the summer months, finding that the mean thaw depth increases as the summer progresses, while the variance and range of thaw depths decrease. This suggests that the system becomes more stable throughout the summer, with the thaw depth plateauing around August and September. Areas with high snow depth at the beginning of the thaw season are likely to initially thaw less than areas with lower snow depth, but this trend becomes less pronounced over time. An interesting artifact of the data was the slight increase in thaw depth at the highest snow depths for the later months, but this finding needs to be confirmed with a geologist.

6 References

1. Source code for `plots` and `utils` modules: https://github.com/pmhalvor/GEO4460/lab/GIS4
2. Sample data for this task was prepared by the Luc Girod for the course GEO4460 (direct download via UiO Canvas)

7 Appendix

7.1 A.1 Raw measurements residual plots

We wanted to see if the residual plots for the raw measurements also curve upwards at the highest snow depths for the later months. The plot below confirms this for the raw measurements, as mentioned in the analysis section.

```
[7]: _ = plot_snow_melt_regression_lines(snow_melt_raw)
```
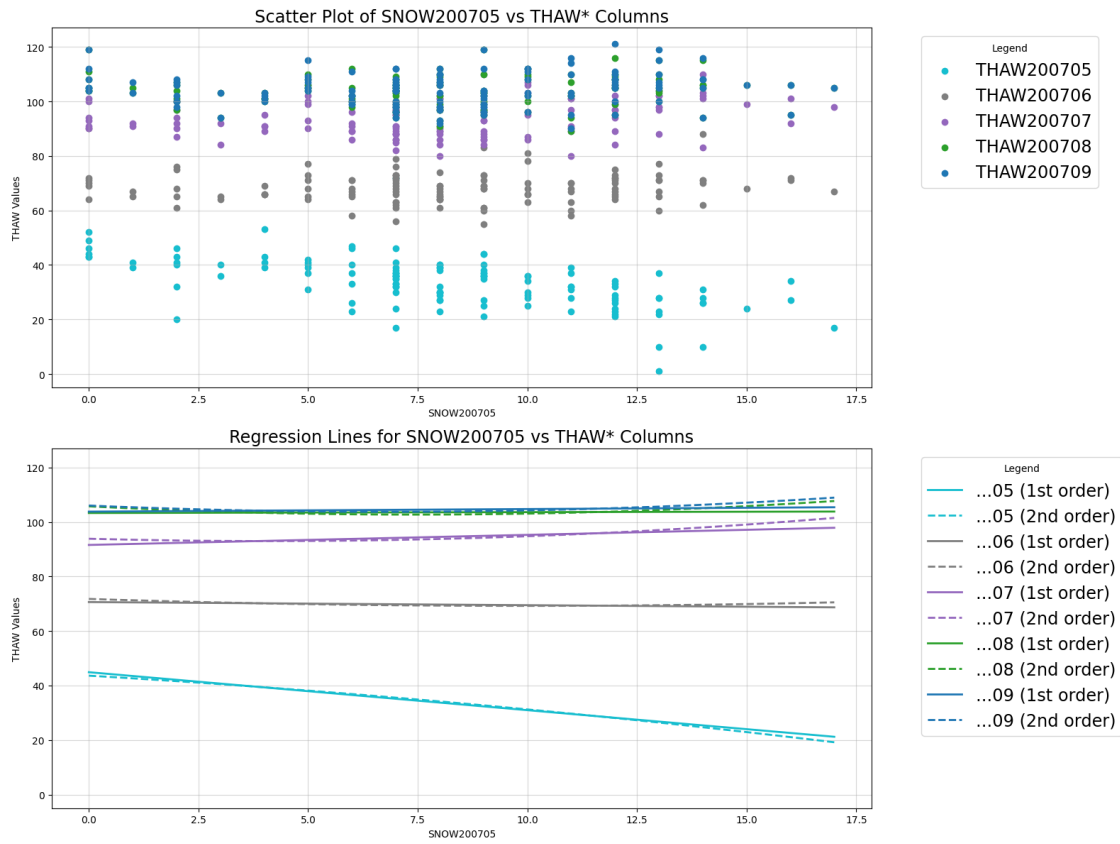
```
<Figure size 1200x800 with 0 Axes>
```


Figure 6: Residual plots for the raw measurements. The residuals are shown in the scatter plot, while the regression lines are shown in the line plot.