Mahmoud Omer Mahmoud Awadallah

# Comparison between the Green and the Red LiDAR terrain models in flood inundation estimations

June 2021

NTNU
Norwegian University of Science and Technology

Master's thesis
2021
Mahmoud Omer Mahmoud Awadallah

Master's thesis

NTNU
Norwegian University of
Science and Technology
Faculty of Engineering
Department of Civil and Environmental Engineering

NTNU
Norwegian University of Science and Technology

Norwegian University of
Science and Technology

# Comparison between the Green and the Red LiDAR terrain models in flood inundation estimations

**Mahmoud Omer Mahmoud Awadallah**

Hydropower Development
Submission date: June 2021
Supervisor: Knut Alfredsen
Co-supervisor: Ana Juárez

Norwegian University of Science and Technology
Department of Civil and Environmental Engineering

## Preface

This thesis is a final product of the course TVM4915 Hydropower Development, Master's Thesis. It is submitted to the Norwegian University of Science and Technology (NTNU). The topic of the thesis is a comparison between the Green and the Red LiDAR terrain models in flood inundation estimation.

I would like to express my gratefulness to my supervisors' professor Knut Tore Alfredsen and the Ph.D. candidate Ana Juárez Gómez. Both have given me feedback and advice, especially on topics related to flood modeling and river analysis. Thank you for your inspiration and guidance during the writing process and great support throughout the master´s thesis semester.

I would also like to thank:

- Nuha Mahdi, my fiancé so far and future wife very soon. Special thanks to her support, love, and belief in me. I would not reach this stage without you being beside me.
- Elhadi Mohsen for his supportive ideas and motivation in coding and thesis structure.
- The HPD classmates for the pleasant master journey and the great time we spent together.
- My family in Sudan for the unlimited support and motivation.
- To the Sudanese people in my country who are suffering to get on their feet after December's revolution.
- To everyone who helped in making this possible.

Trondheim, June 10th, 2021

Mahmoud Awadallah

## Abbreviations

| | |
|---|---|
| LiDAR | Light Detection And Ranging |
| RL | Red Light Detection And Ranging |
| GL | Green Light Detection And Ranging |
| DEM | Digital Elevation Model |
| GDEM | Global Digital Elevation Model |
| WSE | Water Surface Elevation |
| USGS | United States Geological Survey |
| UTM | Universal Transverse Mercator |
| SRTM | Shuttle Radar Topography Mission |
| QM | Mean flood scenario discharge |
| Q10 | 10-years flood scenario discharge |
| Q20 | 20-years flood scenario discharge |
| Q50 | 50-years flood scenario discharge |
| Q100 | 100-years flood scenario discharge |
| Q200 | 200-years flood scenario discharge |
| Q500 | 500-years flood scanrio discharge |
| LLærdal | Lower Lærdal |
| LSurna | Lower Surna |
| ULærdal | Upper Lærdal |
| USurna | Upper Surna |
| NVE | Norges Vassdrags- og Energidirektorat |

# Table of Contents

## Thesis structure

This thesis has been written in paper format, an untraditional format for the master's thesis. The paper is planned to be submitted to the Journal of Flood Risk Management. Therefore, a manuscript of the paper ("Comparison of the Green LiDAR and the Red LiDAR terrain model in flood inundation estimation") will be the main content of the thesis.

The study has included 11 sites in Norway where good quality data of both LiDARs exist. The thesis work included hydraulic simulations by HEC-RAS 6.0, which was developed by the US Army Corps of Engineers, ArcGIS models, Python coding (to automate the ArcGIS processes), and R Language coding (to post-process the data). Shown in the manuscript is only the final developed methodologies, and the initial attempts that led to these final methodologies have been included in the appendix section. Also included in the manuscript are only selected results for the tested sites. Others were included in the appendices too, and the whole set of the results could be found in the digital storages of the thesis work.

The main content of the manuscript includes:

- Chapter1: contains a literature review of the use of digital elevation models in flood inundation estimations, with more focus on LiDAR data and the issues of the Red LiDAR data.
- Chapter2: presents a brief description of the sites used, their locations, and the relevant specifications.
- Chapter3: presents the developed methodologies and assumptions for the hydraulic simulations and the terrain analysis.
- Chapter4: presents some of the interesting results obtained from the study.
- Chapter5: discusses the obtained results with detailed investigations and connects the findings with the previous relevant studies.
- Chapter6: contains conclusions and recommendations for the usage of the Red LiDAR data with respect to the different reach's specifications.

The appendix section contains:

- Appendix A: presents the developed methodologies used throughout this study. These methodologies include estimating the LiDAR inundatation error in transects, the missed LiDAR volume in transects, bank's slope using "the transect length" concept, and bank's slope using "the point elevation" concept. Each methodology consists of the ModelBuilder model and the python code for the automation.
- Appendix B: presents the complete sets of the flood inundations and the correlation relationships results for two sites. Also, the bank's slope relationship with the LiDAR inundation error in transects for all the sites are included.
- Appendix C: presents a table contains the name of the projects of the LiDAR data used in this study.
- Appendix D: contains a correlation matrix (plot) of the terrain parameters with the LiDAR error to test the correlation of the reach's parameters with the total LiDAR inundation error for the sites.

# Comparison between the Green and the Red LiDAR terrain models in flood inundation estimations

**Abstract:** LiDAR data has revolutionized the availability of digital elevation models, basis to generate flood inundations. Red LiDAR is the common methodology for scanning terrain, but, for hydraulic applications, it cannot penetrate water and thereby lacks river bathymetry. Therefore, using the Red LiDAR terrain for producing flood inundation maps will be accompanied by misleading results. On the other hand, Green LiDAR can penetrate water and provide detailed bathymetry. This study has used the 2D-hydraulic flood simulation; to compare the performances of both LiDAR data in 11 sites in Norway with good available data. Investigations of the associated geomorphological features of the sites were carried out to link the differences in inundations to terrain parameters. It is found that the inundation error continuously decreases as the flood return period decreases in reaches where few flood protection embankments are present. On the other hand, in reaches where substantial flood embankments are built, the error increases until the flood levels overtop the embankments before it returns to a decreasing trend. Moreover, the level of the inundation error was found to be positively correlated with the percentage of the protection coverage and negatively correlated with the bank's slopes. A correction was implemented by subtracting the discharge at time of the Red LiDAR flight from the flood discharges, and we found this to be difficult in rivers with flood protection works. It is recommended that the use of the Red LiDAR terrains for flood inundation estimations for flat bank reaches subjected to significant flood protection should be cautious. More caution should be provided the larger the extent of the flood protection works exist in the reach.

Keywords: Red LiDAR, Green LiDAR, flood, inundation error, DEM

## 1 Introduction

Floods are one of the natural disasters that humans have suffered from throughout their existence. In Europe, river floods are the most common natural catastrophes, resulting in significant economic loss through direct damages to properties, infrastructures, and agricultural lands, and indirect losses within the affected areas (Ciscar et al., 2011). Moreover, the situation is projected to worsen more as studies expect global warming to increase the frequency and the intensity of the extreme precipitation events in addition to flood levels (Blöschl et al., 2017; Christensen & Christensen, 2003; Frei et al., 2006; Lawrence, 2020).

Floods, in general, can be categorized in two forms: The first is Pluvial floods, which are floods mainly generated by excess rainfall and topography where inundation occurs due to exceedance of the infiltration capacity (Apel et al., 2016; Rözer et al., 2016; Tanaka et al., 2020). Many studies have investigated the severity of this kind of flood (Jiang et al., 2018; Maksimović et al., 2009; Zhou et al., 2012). The other type of flood is the fluvial flood, which is dependent on an overflowing water body. This type of flood happens when the water level in the river, lake, or reservoir rises to overflow to the adjacent floodplains. The causes of this rise can be a result of excess rainfall in the upstream part, snow melt, from landslides in the lake or reservoir, or due to river blockages (Zurich, 2020). The simulation of fluvial floods is maintained by hydraulic simulations where the terrain model, or the digital elevation model, is one of its backbone inputs.

The U.S. Geological Survey (United States Geological Survey USGS) has defined the Digital elevation models (DEMs) as regularly spaced arrays of elevation values that are referenced horizontally to a specific geographic coordinate system or to a commonly Universal Transverse Mercator (UTM) projection. DEMs can be generated from many sources such as ground-based surveys, digitizing hardcopy existing topographic maps, or utilizing remote sensing. Remotely sensed DEMs, mainly, have been the most used type of terrain models for flood studies, and Shuttle Radar Topography Mission

(SRTM), among other remote sensed terrain models, is the most used DEM type, thanks to its availability, acceptable resolution, and accuracy (Muhadi et al., 2020). Accurate DEMs have vital importance in supporting the modeling of the environmental process (Jarihani et al., 2015), and for floods especially, geospatial forms such as digital elevation models have contributed positively to flood studies credits to their topographic information (Hafezi et al., 2018; Wang et al., 2018). Many flood studies have been carried out using the SRTM digital elevation models (Azizian & Koohi, 2021; Delaney & Evans, 2015; Jakovljević & Govedarica, 2019; Jarihani et al., 2015; Kumar & Acharya, 2016; Maruti et al., 2018; Zhang, 2020). The availability of the SRTM DEMs even in sparse data regions has contributed to its widespread use. However, SRTM has significant limitations in its usage in flood models. The inability of SRTM to represent the complex city's landscapes with coarse resolution (30 m or 90 m) and the failure of the originator radar to penetrate vegetation (Muhadi et al., 2020) have set SRTM as not the most favorable terrain model to be used. These limitations have inspired other remote sensing technology to be used that could tackle SRTM flaws, and one of the recent emerging technology is the Light Detection And Ranging technology (LiDAR).

LiDAR (Light Detection And Ranging) is a remote sensing technology that enables rapid and accurate development of actual 3-dimensional images. The technology calculates the time that an emitted laser signal travels and the reflected returned signal (time of flight scanner) or calculating the phase difference between the laser signal (Lefsky et al., 2002). The generated DEMs from LiDAR technology have many advantages over the other sources of DEM. LiDAR data could be acquired during the day, night, and even during cloudy conditions (Dowman, 2004; Hodgson et al., 2003). Also, it can penetrate the vegetative areas and the urban structures such as the buildings, the bridges, and all the manmade features. This would result in a more accurate representation of the ground that is difficult to be achieved through other remote sensing technology. In addition to that, the accuracy of the LiDAR generated DEMs can reach down to centimeters accuracies, which has expanded their usage in flood modeling (Muhadi et al., 2020).

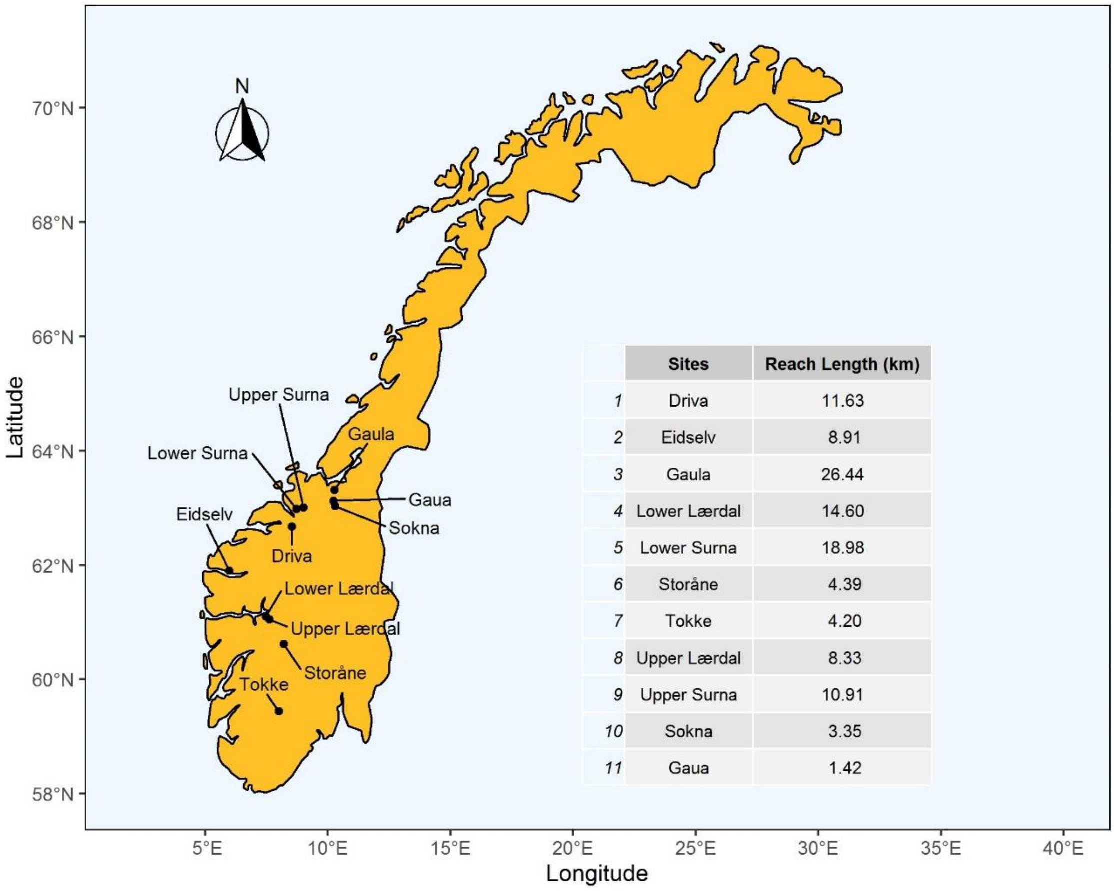
Many flood studies have used the LiDAR DEMs as a benchmark digital elevation model to estimate the error resulting from using coarse DEMs in producing flood inundation maps. For instance, flood inundation maps generated by global digital elevation models (GDEMs), such as SRTM, have shown to perform less accurately than the results obtained by LiDAR DEMs (Bhuyian & Kalyanapu, 2018; McClean et al., 2020; Muthusamy et al., 2021). The studies revealed an overestimation of the flood extent when coarse resolutions are used, which will provide misleading inundation maps, especially for urban areas. The main reason for this increase is the loss of the river channel definition when a coarse DEM is used, and more amplification of the flood extent will be expected when the resolution of the DEM exceeds the width of the river, which eventually leads to a higher inundated area (Bhuyian & Kalyanapu, 2018; Muthusamy et al., 2021). Therefore, the use of the LiDAR DEM has been a significant advancement in lessening flood inundations' errors associated with rough DEMs. However, even with the ultimate horizontal capabilities, the traditional LiDAR DEMs are responsible for generating erroneous flood inundation extent attributed to their misrepresentation of the river bathymetry. Conventional LiDAR, or what it is widely known as Red LiDAR (RL), is obtained by infrared laser that is unable to capture the underwater geometry since the laser is absorbed by the water surface (Casas et al., 2006). Many studies have inspected this type of error by implementing measured field bathymetry to the Red LiDAR DEMs (Bures et al., 2019; Choné et al., 2018; Dey et al., 2019; Reil et al., 2018). They have all concluded that omitting the bathymetric data in using Red LiDAR DEMs will result in an overestimation in the simulated inundations. However, the recent advancement of the LiDAR technology has emerged as another kind of LiDAR that tackles the Red LiDAR issues. The Green (GL), or the bathymetric, LiDAR, unlike the Red LiDAR, can penetrate the water since it uses blue/green laser beams (Hilldale & Raff, 2008; Irish & Lillycrop, 1999; Kinzel et al., 2013; Mandlburger et al., 2015). Therefore, with the expected increase of the availability of the Green LiDAR data, this source of bathymetric data would be the favored among its peers since it provides more extensive coverages of the rivers in considerably less amount of time and effort. However, no studies have conducted an explicit comparison between the Green and the Red LiDAR DEMs regarding the flood estimation. Presently, 80% of Norway is covered by the Red LiDAR DEM and a full coverage is expected by 2023 (Breili et al., 2019). Therefore, since the Red LiDAR is far more available than the Green LiDAR, it is interesting compare their performance in estimating the flood inundations. Moreover, in all the studies that have addressed the error associated with the missing or inadequate bathymetry, no study has linked the error in inundation to the geomorphologic features of the river. This paper seeks to answer the following research questions:

1. How to quantify the difference in inundations in using Green and Red LiDAR DEMs across different flood scenarios (Mean flood, 10-years, 20-years, 50-years, 100-years, 200-years, and 500-years floods)?
2. Can Red LiDAR be used as a basis for flood inundation studies?
3. What is the relationship between the error in inundation between red and green lidar based DEM and the geomorphological features of the river? Can this be used to inform users on sites were red lidar could be used in flood inundation studies?

# 2 Data

## Study sites and the LiDAR data

Eleven sites in Norway were selected in this study based on the availability of both Green and Red LiDAR data in each reach. The sites are in five counties (Three sites in Trøndelag, four sites in Møre og Romsdal, two sites in Vestland, and one site in each of Viken and Vestfold og Telemark Municipalities) were used in this study. Figure 1 shows the locations of the sites with respect to Norway provided with information on their lengths.




| | Sites | Reach Length (km) |
|---|---|---|
| 1 | Driva | 11.63 |
| 2 | Eidselv | 8.91 |
| 3 | Gaula | 26.44 |
| 4 | Lower Lærdal | 14.60 |
| 5 | Lower Surna | 18.98 |
| 6 | Storåne | 4.39 |
| 7 | Tokke | 4.20 |
| 8 | Upper Lærdal | 8.33 |
| 9 | Upper Surna | 10.91 |
| 10 | Sokna | 3.35 |
| 11 | Gaua | 1.42 |

*Figure 1: The reaches locations in Norway with inofrmation of their lengths*

Table 1 shows a general description for the tested sites. The sites cover wide range of river sizes with the Gaula reach having the largest catchment area and reach length of 3086 $km^2$ and 26.44 km respectively, and the smallest catchment area and reach length is found in Gaua with 85 $km^2$ and 1.42 km respectively. All the sites have a good quality LiDAR data with a maximum horizontal resolution of 0.5 m, and 0.25 m resolution representing the majority among the sites. The LiDAR DEMs were generated by the Norwegian Mapping Authority and made available through website *www.høydedata.no* .

*Table 1: Sites' info summary with their LiDAR data properties*

| ***Site*** | ***Catchment Area ($km^2$)*** | ***Mean Discharge ($m^3$/sec)*** | ***RL Resolution (m)*** | ***RL Point density (points/$m^2$)*** | ***RL Flow (m3/s)*** | ***GL Resolution (m)*** | ***GL Point density (points/$m^2$)*** |
|---|---|---|---|---|---|---|---|
| Driva | 2436 | 63.6 | 0.5 | 2 | 74 | 0.25 | 4 |
| Eidselva | 386 | 23.4 | 0.25 | 5 | 17.6 | 0.25 | 4 |
| Gaula | 3086 | 83.3 | 0.25 | 6 | 146 | 0.25 | 5 |
| Lower Lærdal | 994 | 30.7 | 0.5 | 2 | 14 | 0.25 | 5 |
| Lower Surna | 910 | 40.6 | 0.5 | 2 | 20 | 0.5 | *NA* |
| Storåne[1] | 770 | 24.5 | 0.25 | 5 | 6.8 | 0.2 | 20 |
| | | | | | 35.9 | | |
| Tokke | 2332 | 89.5 | 0.25 | 5 | 22.9 | 0.25 | 20 |
| Upper Lærdal[2] | 750 | 23.0 | 0.25 | 5 | 26 | 0.25 | 5 |
| | | | 0.5 | 2 | 13 | 0.25 | 5 |
| Upper Surna | 445 | 17.4 | 0.5 | 2 | *NA* | 0.5 | *NA* |
| Sokna | 564 | 13.0 | 0.25 | 5 | 15 | 0.25 | 5 |
| Gaua | 84.6 | 2.0 | 0.25 | 6 | 1.5 | 0.25 | 5 |

*1 has two LiDAR flow values because of powerplant outlet in the mid of the reach.*
*2 has two different RL DEMs cover the reach.*
*RL and GL denote for Red LiDAR and Green LiDAR respectively.*

## Flood data

The flood scenarios for the sites used in the simulation are shown in Table 2. For eight sites, the simulated discharges were provided by Norwegian Water Resources and Energy Directorate (NVE), while for the three remaining, the values were taken from the NVE website, www.nevina.no, which provides unregulated flood scenarios for Norwegian rivers based on a regional flood frequency analysis (Engeland et al., 2020).

*Table 2: Flood scenarios discharges for the sites (in $m^3/s$)*

| Site | Q M | Q 10 | Q 20 | Q 50 | Q 100 | Q 200 | Q 500 |
|---|---|---|---|---|---|---|---|
| *Driva* | 545 | 725 | 795 | 885 | 960 | 1025 | 1115 |
| *Eidselva* | 66 | 86 | 93 | 101 | 107 | 112 | 118 |
| *Gaula* | 1041 | 1551 | 1800 | 2144 | 2404 | 2685 | 3070 |
| *Lower Lærdal* | 235 | 380 | 470 | 570 | 700 | 800 | 890 |
| *Lower Surna* | 229 | 342 | 391 | 454 | 501 | 549 | 613 |
| *Storåne** | 196 | 290 | 327 | 374 | 410 | 446 | 493 |
| *Tokke* | 204 | 289 | 323 | 366 | 406 | 443 | 492 |
| *Upper Lærdal* | 215 | 310 | 350 | 398 | 452 | 495 | 538 |
| *Upper Surna* | 171 | 230 | 254 | 284 | 306 | 328 | 355 |
| *Sokna** | 125 | 194 | 221 | 257 | 284 | 311 | 347 |
| *Gaua** | 21.9 | 34.5 | 39.5 | 46.1 | 51.1 | 56.2 | 63.1 |

**unregulated flood scenarios provided by the open website www.nevina.no .*

# 3 Methodology

## DEM generation for the LiDAR models

The Green LiDAR data are mainly provided for the river's main channel. Therefore, an integration of the river bathymetry the Green LiDAR measurements with the floodplain extent from the Red LiDAR data was done to form a complete LiDAR model of the river and the floodplains. The river's mainstream in the full Green LiDAR model has the actual bathymetry of the river, while in the Red LiDAR model, it has the water surface elevation when the flight measurement was taken. However, they have both the same extent for the floodplains, taken from the Red LiDAR measurement, since the Red LiDAR has a broader coverage than the Green LiDAR. Figure 2 shows an illustration of both LiDAR models.

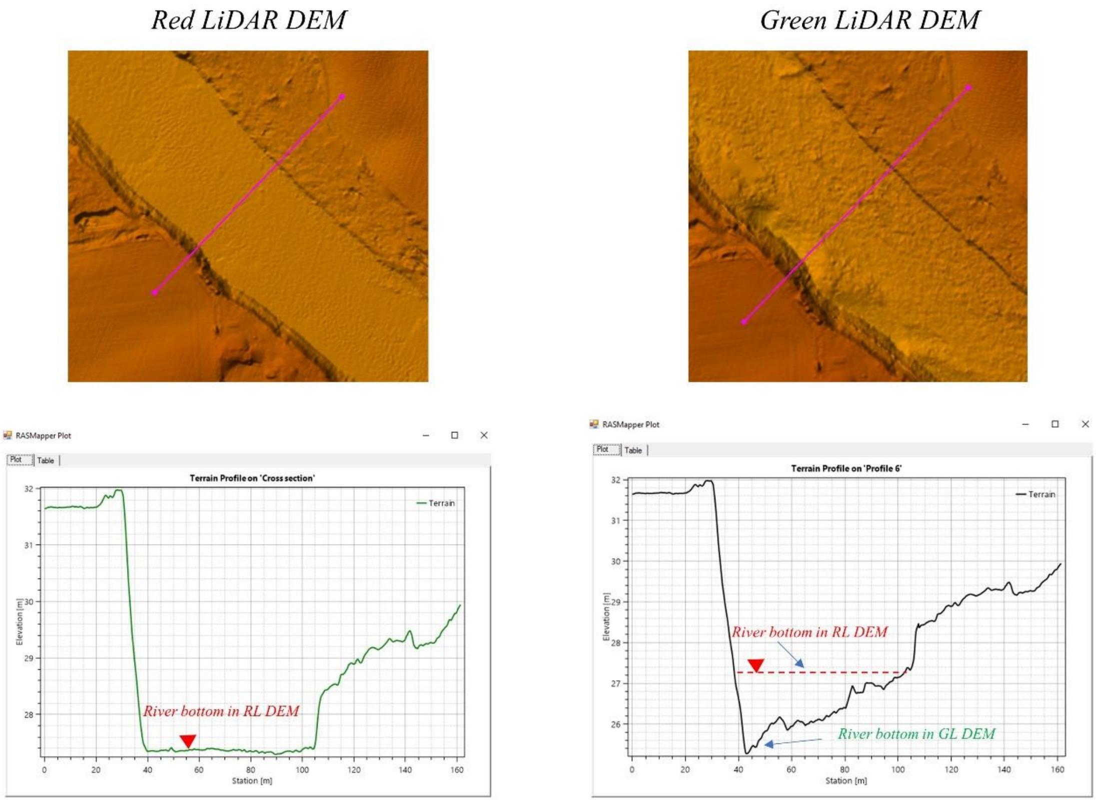



*Figure 2: Cross-sectional view of the Red LiDAR and Green LiDAR DEMs.*

## Hydraulic Simulation

2D flow simulations were carried out for sites using the Hydrologic Engineering Center's River Analysis System (HEC-RAS 6.0) which was developed by the US Army Corps of Engineers (The US Army Corps of Engineers, 2021). The 2D flow equation of the diffusive wave was used which provides stable and speedy simulations. As the steady state model has been used widely to undertake flood hazard mapping (Bradbrook et al., 2004; Cook & Merwade, 2009), an unsteady flow simulation with constant peak discharge was used to obtain the flood inundations in this study.

In order to ensure a common base for comparison between the sites, all the geometrical hydraulic parameters were unified to eliminate differences in inundation due to such differences. A 5 m cell size for the computational mesh was used for all the sites similar to previous studies (Muthusamy et al., 2021). Also, an identical geometry files for the Green and the Red LiDAR models was used to ensure full elimination of inundation' differences due to non-similar mesh layout or boundary's locations for instance. On the other hand, for Manning's coefficient, the approach of studies such as (Jung et al., 2012; Thomas Steven Savage et al., 2016) was followed. A typical two values of Manning's n from the literature were used (Chow, 1959) where, for the river mainstream, a manning of 0.03 was assigned, while for the floodplains, it was given a manning of 0.06 (Muthusamy et al., 2021). A flow hydrograph was used for as the upstream boundary conditions, while for the downstream boundary condition, a normal depth condition was used for all cases with a slope value of 0.001. It is important to mention here that this study aims to check the variation in flood inundations resulting from using the different LiDAR models, therefore the results cannot be used as referenced flood inundations for the sites due to the unification of the manning's n and the downstream boundary conditions.

## Terrain Analysis

One of the objectives of the study is to investigate the link between the error in inundation and some geomorphological indices of each river. The investigated parameters are listed below.

### 3.1.1 Reach's longitudinal slope, sinuosity, and flood wall coverage

The longitudinal slope of the river is one of the key parameters that defines the flow in the open channel flow systems (Chow, 1959). Therefore, it has been considered in the investigations as one of the parameters that could define how the inundation's error progresses. For each river, the parameter is obtained by the elevations of the starting and ending points of the reach divided by the length of the reach. The sinuosity index is also obtained for each site as the ratio of the actual meandered length of the reach to the straight line connecting the starting and ending points of the reach. The percentage of coverage of the reach's banks with the flood protection walls was also estimated. The data of the flood protection works for the Norwegian rivers are openly available from www.nedlasting.nve.no/gis .

### 3.1.2 Missed LiDAR volume

The volume lost by scanning with the red LiDAR was computed as the difference between the Green LiDAR bathymetry and the water surface recorded using the Red LiDAR. The Raster Calculator tool in ArcGIS combined with the Zonal Statistics tool were used to find the total to find the total volume difference between the two rasters.

### 3.1.3 Bank slope

The shape of the riverbank is a factor that could explain the variability in difference in inundation between the Red and Green LiDAR geometries. Unlike the previous parameters, this parameter was computed at a cross-sectional level along the reach. The parameter was extracted by a series of tools using ArcGIS Pro automated in a Python script.

To capture the shape variation of the bank slopes, ten layers were created to cover the confined region between the bottom of the river and the maximum water level resulting from the Q500 scenario. Then, the slope of the bank between each two layers was found by calculating the horizontal distance (Z) of the sloped surface with respect to a unit elevation difference. To calculate Z, the difference in the horizontal distances where the layers intersect the bank and the difference of the elevations of each layer were obtained. The horizontal length L, the normal distance between the river centerline and where the layer intersects the bank, was found by clipping the transects to the boundary of each layer. Then Z could be calculated with the following equation:

$$Z = \frac{L_{i+1} - L_i}{M_{i+1} - M_i}$$

Where Z is the horizontal distance of a unit elevation of the slope, L is the horizontal distance from the river line to where the layer meets the bank, M is the elevation of the layer, and (i) and (i+1) are the lower layer and the upper layer respectively. The bank's slopes between the ten layers were found for the left and the right sides of the river centerline. The selection of which bank slopes to include for each flood scenario was based on the Water Surface Elevation (WSE) of each scenario. The first layer that has an elevation immediately below the WSE of the GL QM scenario was specified as the lowest layer. The highest layer was the layer that has an elevation right higher than the WSE of RL flood scenario. From the selected layers, the mean of the bank slope is computed as well as the standard deviation of the bank's slopes. The horizontal distance of the slope (Z) was converted into degrees afterwards as it confines the slope's variation between 0 to 90 degrees.

## Evaluation of the flood extents

The study targets to investigate the variation in the flood inundations resulting from using the different LiDAR DEMs as terrain in the hydraulic simulations. To address a full investigation, the evaluation was carried out in two different scales: the reach scale and the transect scale. For the reach scale, the error was computed considering the total inundation area of a certain flood scenario. Normalized Error was the parameter implemented to quantify the overestimation in inundation created by RL DEM, and is calculated as follows:

$$Normalized\ Error = \frac{RL\ Inundation}{GL\ Inundation}$$

Where:
RL Inundation – The flood extent for a flood scenario produced by the Red LiDAR model ($m^2$).
GL Inundation – The flood extent for a flood scenario produced by the Green LiDAR model ($m^2$).

Therefore, the closer this parameter to 1, the closer the resulting inundation to the actual flood extent defined by the Green LiDAR which is considered as the ground truth. This parameter was computed for each flood scenario for the site, and a comparison between the errors of the different sites was held. For the transect scale the inundation's error was computed as the Red LiDAR deviation from the Green LiDAR as follows:

$$Inundation\ Error\ (\%) = \frac{RL\ Inundation\ Length - GL\ Inundation\ Length}{GL\ Inundation\ Length} \times 100$$

Where:
RL Inundation Length – the length of the transect covers the inundation from the RL model (m)
GL Inundation Length – the length of the transect covers the inundation from the GL model (m)

Therefore, the smaller this percentage, the closer the RL inundation to the actual inundation

# 4 Results

Figure 3 shows an example of the extracted flood inundations from 2D HEC-RAS using GL and RL DEMs. The inundations belong to Gaula site. It can be seen that the RL Inundation results in a larger flooded area than the GL Inundation. This happens along the river reach. Also, the variation of the flood extent differs from location to another. In some locations, the two extents almost match with minor differences, while in others the variations are substantial.

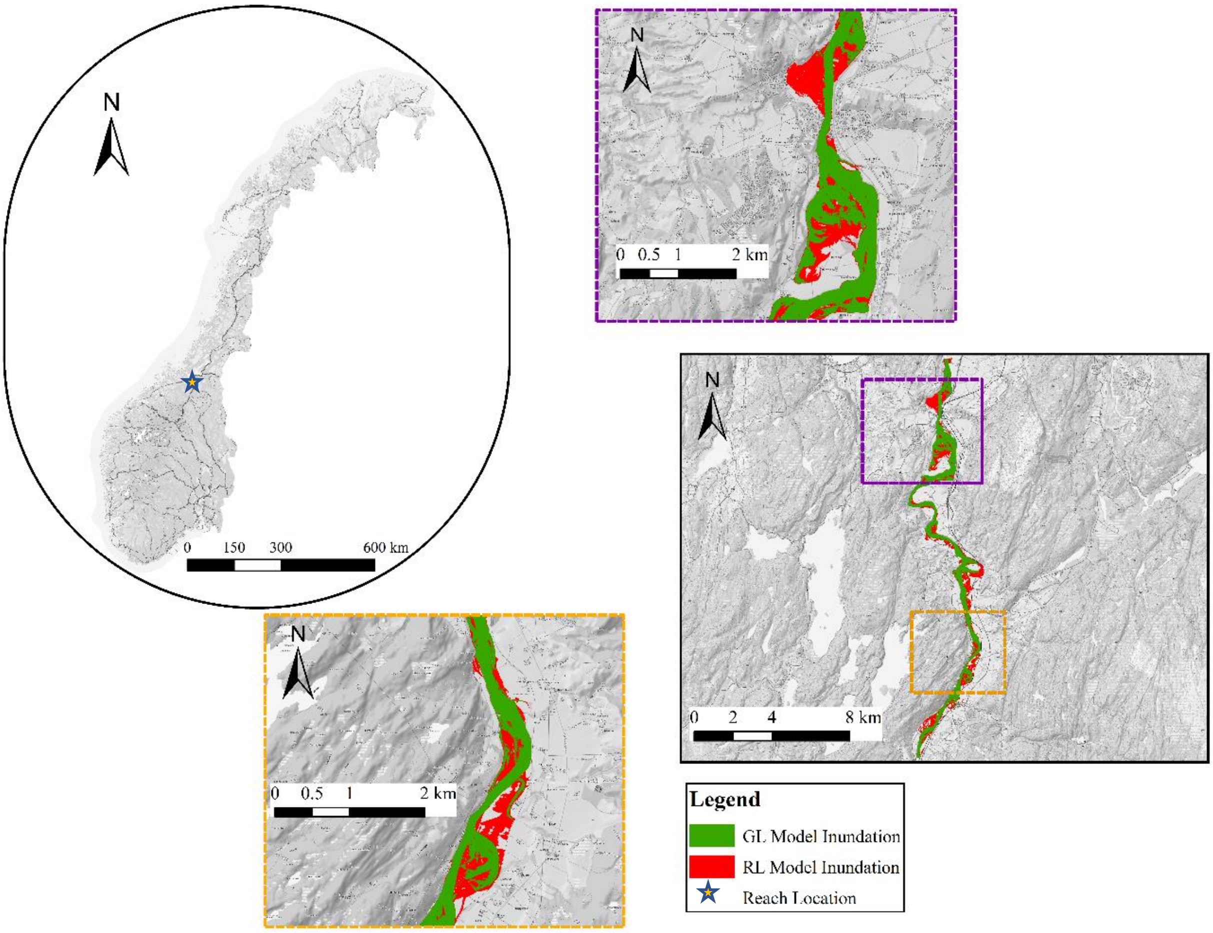



*Figure 3: Flood extent for Q-50 years scenario obtained by HEC-RAS 2D model for Gaula site using different LiDAR DEMs. GL and RL denote Green LiDAR and Red LiDAR respectively.*

are the most protected reaches among the tested ones with coverage of 72 and 62 % respectively, while Storåne and ULærdal have the least protection with 0 and 3 % coverage respectively.

Table 3 shows a summary of the terrain analysis parameters extracted through treating the site as one unite. For the missed LiDAR Volume, it can be seen that Gaula has the largest discrepancy among the sites with 141 $m^3$ missed bathymetry volume per meter length, while Storåne has the lowest value with 4 $m^3$ per meter length. Meanwhile, Gaula shows to have the mildest slope with 0.12 % while the highest slope was found in ULærdal with 1.9%. For the sinuosity, Eidselva shows the highest sinuosity with 1.83 while Gaua and USurna have the lowest with a value of 1.14. For the flood protection coverage, LLærdal and Gaula are the most protected reaches among the tested ones with coverage of 72 and 62 % respectively, while Storåne and ULærdal have the least protection with 0 and 3 % coverage respectively.

Table 3: Terrain analysis outputs for the 11 sites

| Site | Missed LiDAR Volume ($m^3/m$) | Slope (%) | Sinuosity | Flood Protection Coverage (%) | Mean Bank's Slope (Degrees) |
|---|---|---|---|---|---|
| Driva | 17 | 0.29 | 1.37 | 24 | 2.97 |
| Eidselva | 26 | 0.49 | 1.84 | 11 | 9.88 |
| Gaula | 141 | 0.12 | 1.33 | 62 | 1.66 |
| LLærdal | 8 | 0.42 | 1.63 | 72 | 2.15 |
| LSurna | 36 | 0.14 | 1.27 | 25 | 2.41 |
| Storåne | 4 | 0.57 | 1.25 | 0 | 3.25 |
| Tokke | 27 | 0.51 | 1.29 | 15 | 11.24 |
| ULærdal | 5 | 1.90 | 1.35 | 3 | 19.54 |
| USurna | 5 | 0.36 | 1.14 | 20 | 5.82 |
| Sokna | 15 | 0.92 | 1.15 | 39 | 15.01 |
| Gaua | 5 | 0.46 | 1.14 | 15 | 4.76 |

The bank's slope distribution is shown in Figure 4 (a) as density plots for the sites. In general, the bank slope distributions for the sites appear as skewed distributions. LLærdal has the largest proportions of cross sections with bank's slope less than 3 degrees, followed by Gaula and LSurna. Storåne and Gaua have their peaks densities at values less than 3 degrees too, but at least one-third of the slopes is greater. On the other hand, ULærdal, Sokna, and Tokke appears as the sites with the largest percentage of bank's slopes larger than 10 degrees. Two sites fall outside the above categories, Eidselva and USurna which both show almost symmetrical bank slope distributions. USurna appears with a narrow peak distribution while Eidselva shows a broad peak range between 3 and 30 degrees. The longitudinal profiles for the sites are shown in Figure 4 (b). Reaffirming what was presented in Table 3, ULærdal is shown to be the clearly steepest study site compared to the others, while LSurna and Gaula is the sites with mildest slope. Figure 4 (c) shows the inundation errors for all the sites with respect to the flood scenarios return period (T). The pattern of the error can be categorized in two types. The first which has a continuous descending pattern where the shorter the return period the higher is the Normalized Error. This type of error distribution is seen in Gaua, Tokke, Storåne, and ULærdal. The second category appears in Driva, Eidselva, Gaula, USurna, and LSurna where the maximum error is not in the flood scenario with the lowest return period. The common feature of the latter category that it reaches a single peak at a certain return period before it shows the same descending trend as the first category. However, LLærdal stands as an outlier with two error's peaks.

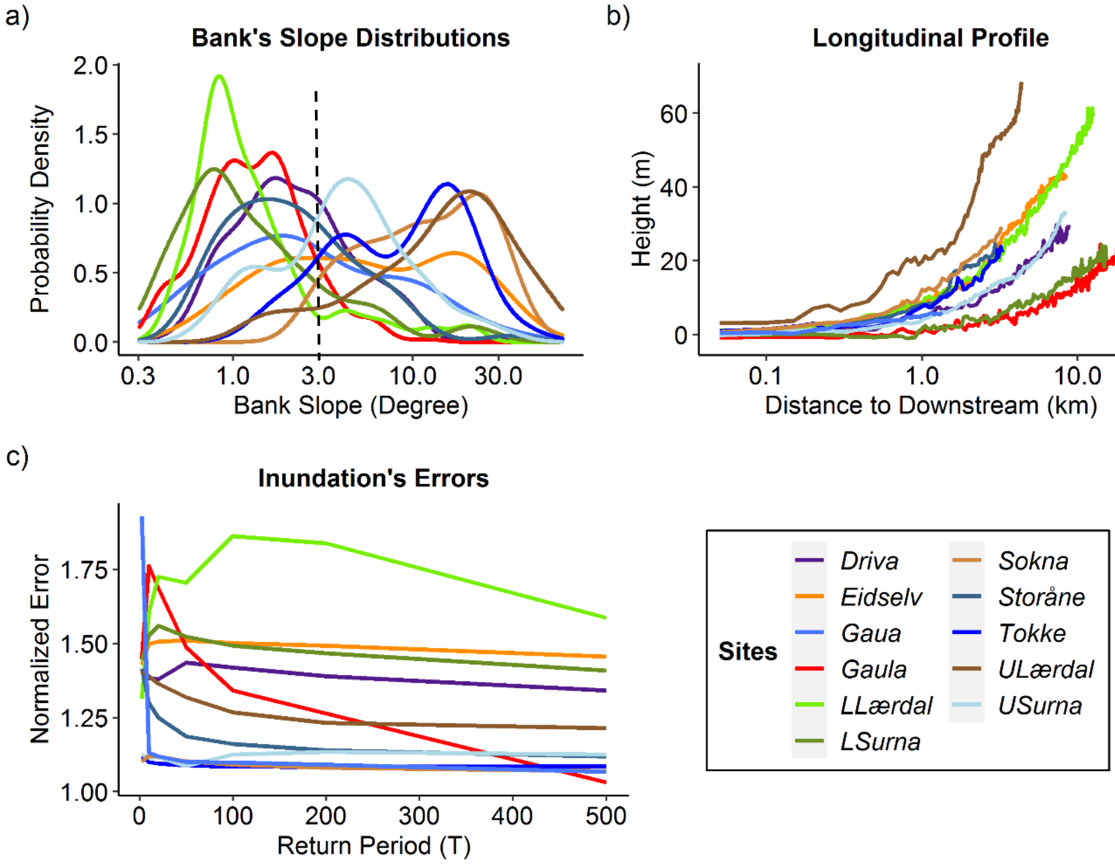



*Figure 4: The bank's slope's distribution, the longitudinal profile and the inundation's error for the 11 sites.*

Figure 5, Figure 6, and Figure 7 show the relationship between the error in inundation on a cross-sectional scale in relation to bank slopes for the slopes for the different flood scenarios. The figures represent sites Eidselva, Gaula, and ULærdal respectively. Those sites cover the different variations in the longitudinal and bank's slopes. In general, the cross-sections that have a lower mean bank's slope angle produce higher inundation errors. Meanwhile, the cross-sections associated with low standard deviation of the bank's slope angles result in higher inundation's errors too.

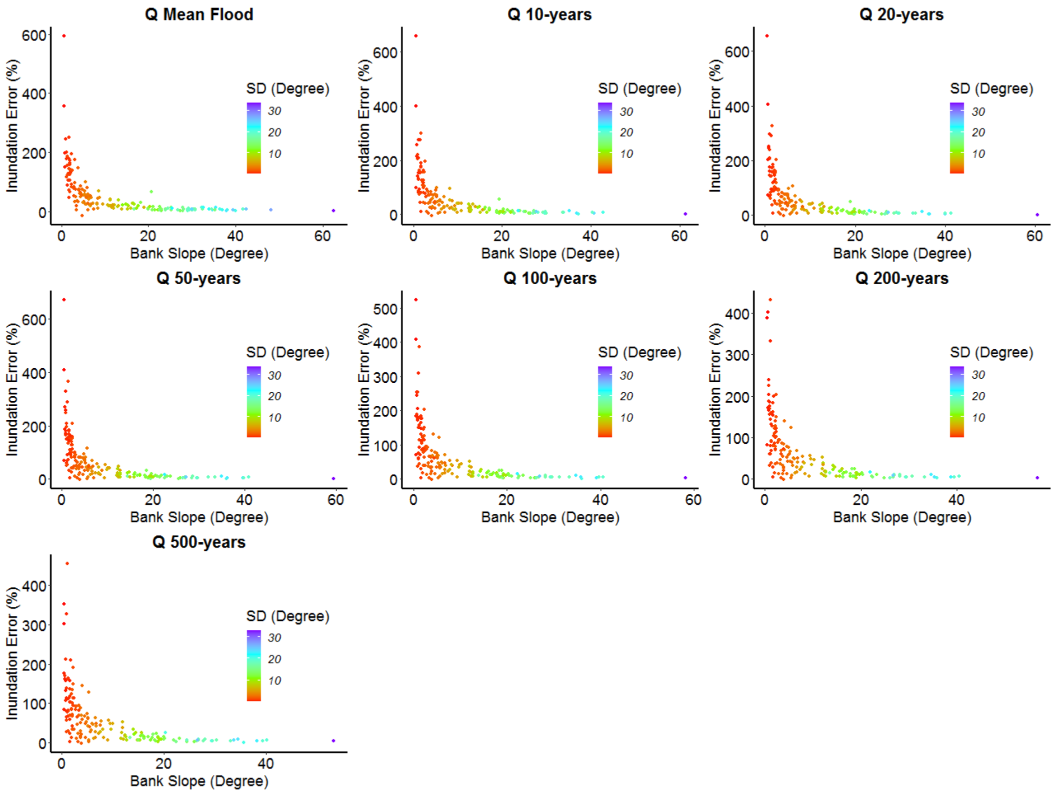



*Figure 5: The relationship between the inundation error in transects and the correspondence mean bank's slope in Eidselva. SD is the standard deviation of the bank's slope at a transect.*

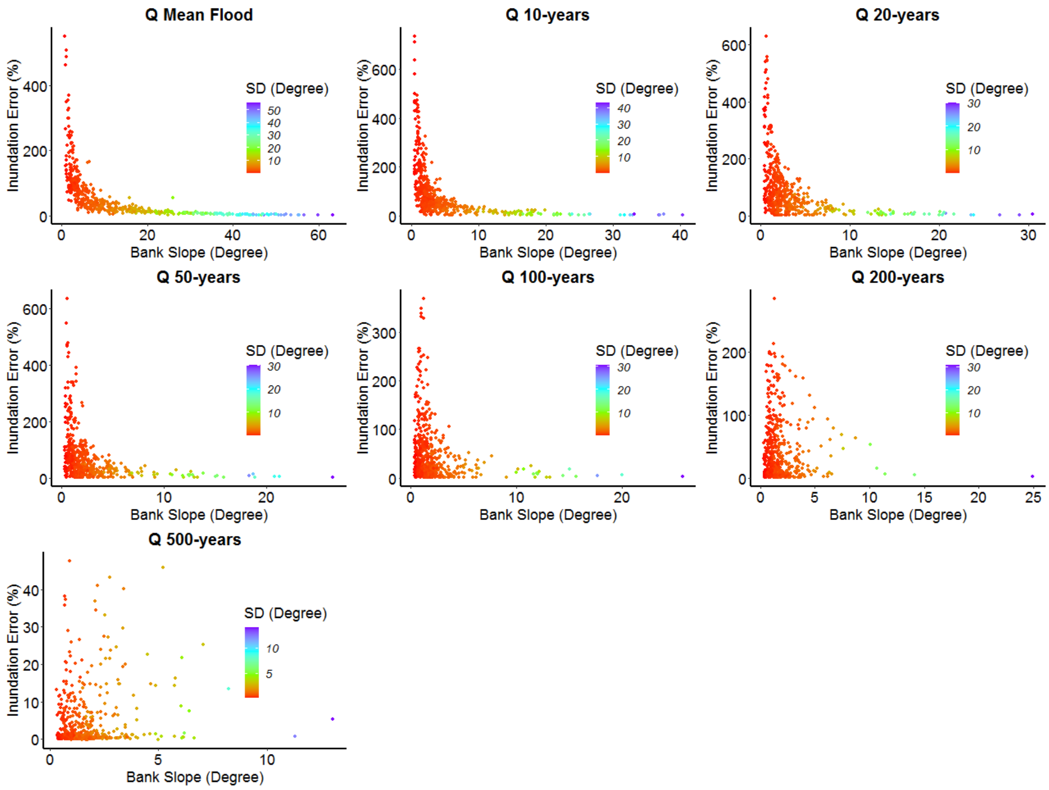



*Figure 6: The relationship between the inundation error in transects and the correspondence mean bank's slope in Gaula. SD is the standard deviation of the bank's slope at a transect.*

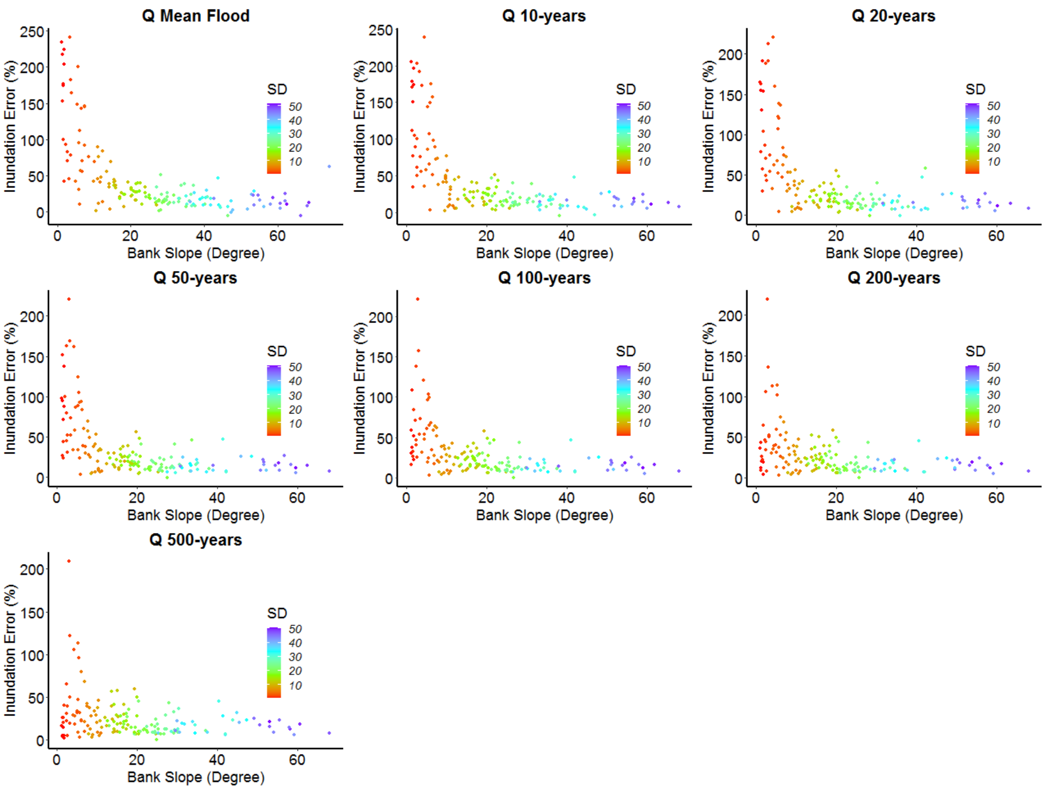



*Figure 7: The relationship between the inundation error in transects and the correspondence mean bank's slope in ULærdal. SD is the standard deviation of the bank's slope at a transect.*

# 5 Discussion

Generally, investigating the error sources in flood estimation models has been a great concern in flood studies(Hall & Solomatine, 2008; Merz & Thieken, 2005). Besides, studying the overestimation resulting from the absence of bathymetry data has been explored too. However, no study has compared the performances of estimating the flood inundations using bathymetric LiDAR (GL) and topographic LiDAR (RL). This study investigated the differences in inundations of different flood scenarios resulting from using the different LiDAR DEMs in several Norwegian rivers of varying sizes and complexities.

## The overestimation in inundation created by the RL DEMs

An overestimation of the flood inundation was detected when using the Red LiDAR DEMs compared to the inundations from the Green LiDAR DEMs which could be seen in the example presented in Figure 3. Such an overestimation was reported in the studies that tested the use of the Red LiDAR DEM against referenced DEMs obtained by field measurements (Bures et al., 2019; Choné et al., 2018). The magnitude of this overestimation varies with changing the flood scenario, and this was also presented by (Bures et al., 2019) in their studies.

## Shape of the inundation’s error curve

Figure 4 (c) shows shape of inundation’s error throughout the flood scenarios in different sites that have diverse river’s features as it is seen in Figure 4 (a), Figure 4 (b), and Table 3.The continuous decreasing trend of the error has been detected in the sites that has the least coverage of the flood protection wall such as Storåne, Tokke, ULærdal, and Gaua where minimal flood protection works has been subjected as it is shown in Table 3. On the other hand, the category that groups the sites that are characterized with the peaking error’s trend agree in having significant flood protection works with LLærdal and Gaula pronounce as the tops. Figure 8 illustrates an example of this embankment effect. Since the bottom of the river is elevated in RL model from where it is been located, the flood walls will be overtopped in the RL model before it does in the GL model. This leads the error in inundation to peak until a larger scenario creates an actual overtopping for the embankment, and, therefore, the error is back to be minimal.

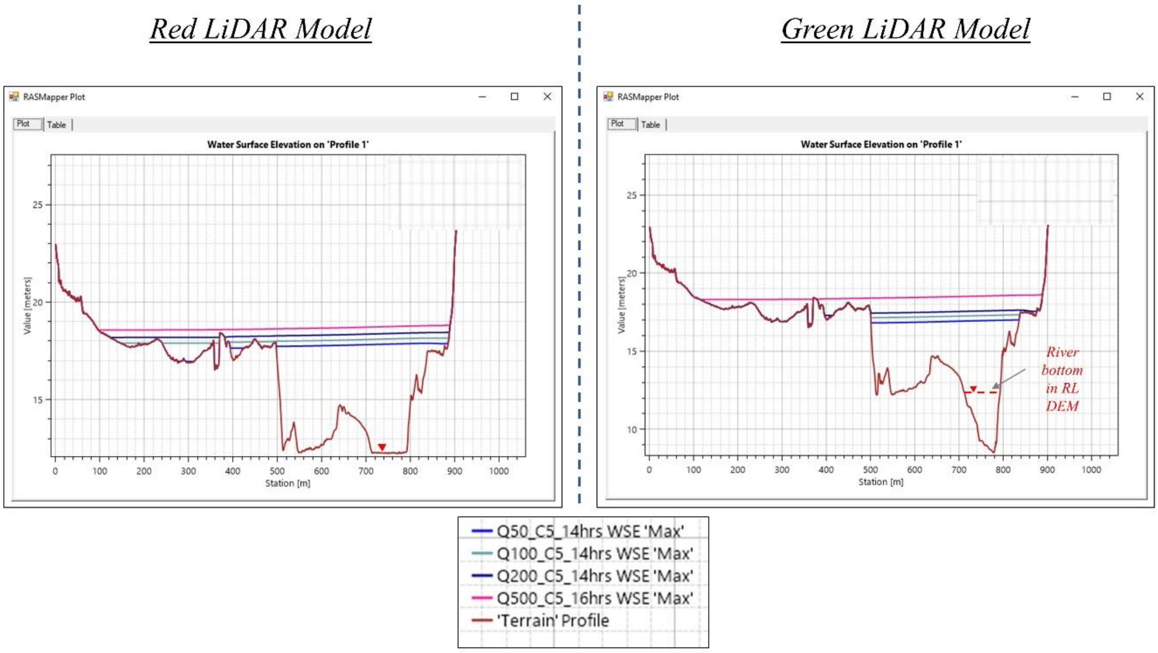



*Figure 8: The water surface elevations for the flood levels Q50, Q100, Q200, and Q500, obtained by 2D HEC-RAS model, in cross section in Gaula site, using the different LiDAR DEMs. RL denotes for Red LiDAR.*

## Level of the inundation’s error curve

In addition to the error’s shape, we can see the sites have not shown the same inundation’s error at each flood scenario, rather, they appear in different levels as shown in Figure 4. To investigate these levels, a correlation test was carried out for the terrain parameters in Table 3 against the mean of the inundation errors of all the scenarios. The mean error was found to have high positive correlations with the sinuosity (0.72) and the flood protection coverage (0.5), and negative correlation with the bank’s slope (0.44). Since the significance of the bank’s slope will be smoothened when the mean slope is only considered rather than the distribution, this parameter is believed to have higher significance. For instance, LLærdal appears to have the highest overall error levels among the other tested sites, and meanwhile it is the site featured with the highest percentage of flat cross-sections as it is shown in Figure 4. On the contrary, the sites that produce the least overall error levels, such as Sokna, Tokke, and ULærdal, were associated with higher percentages of steep slopes than flat banks in their reach. However, some sites were not following the proposed trend. For instance, Storåne even though it is considered among the sites with low error’s levels, it has high percentages of flat banks. However, Storåne site is a steep reach and has a zero coverage of flood protections, which could explain why it has low error's level.

## Inundation's error at cross-sectional scales

The total error that develops at the reach's scale is an aggregation of the error's behavior at the cross-section level, and therefore, understanding the error at this level is equally important. Shown in Figure 5, Figure 6, and Figure 7, the cross sections that have low mean bank's slope's, i.e. flat cross section, is associated with high inundation's error. Meanwhile, the cross-sections that have low standard deviation of the slope angle produce high error as well. This is compatible with the fact that the flatter is bank's slope, the more chance of higher inundation error to happen. The reason for that is because in flat banks, a small overestimation of the flood level will inundate larger area than in steep banks. Therefore, since the use of RL will be associated with erroneously flood level increase, the error in flat cross-sections will be more pronounced. Similarly, a bank slope of a low standard deviation implies a wide cross-section with considerable amount of flat parts. In such cross-sections, the wider the cross-section the higher chance of high inundation error presence. However, the dependency of the cross-sectional inundation error on the bank's slope diminishes the higher the flood scenario is. This could be explained in Figure 8 where at the high scenarios, the flood levels at both LiDAR models almost coincide resulting in the same flooded floodplains.

## Flood model correction

A correction of flood models were proposed by (Bradbrook et al., 2004; Choné et al., 2018) in situations where no channel data are available. The studies have concluded that a better flood estimation could be achieved if the discharge at the time of flight with the RL was subtracted from the flood magnitude. The suggested approach was tested in LLærdal and Tokke reaches and Table 4 shows the obtained improvements. It can be seen that the reductions in errors in Tokke site have shown to be higher than LLærdal in all the scenarios. The study of Choné et al. (2018) has tested this approach with the respect to flood level estimation and found that the approach has failed to produce satisfactory results when it is used in flat to gentle longitudinal slopes (flatter than 0.14%). However, since our comparison concerns the flood inundation, the tested improvements in the inundation area was found to be unsatisfactory even though the reaches were considerable steeper as shown in Table 3. Since the area inundated by a certain flood level will be highly dependent on bank's slope, it can be concluded that the use of this approach for flood inundation estimation should be cautious. Also, from Table 4, it can be seen that the improvements obtained in Tokke were higher than the obtained in LLærdal. The difference in inundation could be explained by the substantial difference in flood protection coverage between the two sites, as it is shown in Table 3. A similar finding was also concluded by Bradbrook et al. (2004) where it stated that this approach of correction fails to provide good results in reaches with considerable flood defense works.

*Table 4: The reduction of inundation error with LiDAR discharge subtraction (in percent).*

| | **Reduction (%)** | |
|---|---|---|
| **Return Period (T)** | **LLærdal** | **Tokke** |
| *2.33* | 10 | 11 |
| *10* | 11 | 13 |
| *20* | 5 | 8 |
| *50* | 6 | 8 |
| *100* | 7 | 8 |
| *200* | 4 | 9 |
| *500* | 6 | 9 |

# 6 Conclusion

In this study, a comparison between the Red and Green LiDAR terrain models has been conducted to evaluate their potential in estimating flood's inundations. The study included eleven sites in Norway where good quality data of both LiDAR models exist. An analysis of how the inundation's errors develops throughout the various reach sizes was carried out. In general, the shape of the inundation errors was found to follow two patterns, the continuously decreasing error pattern and the decreasing with middle peaking pattern. The error patterns were observed to be dependent on the amount of the flood protections implemented at each reach. The excessive introduction of the flood walls in the reach tends to create a peaking pattern in the inundation's errors, and this makes the use of the Red LiDAR models in such reaches misleading. Moreover, the level of the inundation error was found to be correlated with the sinuosity, flood protection, and percentages of the flat bank's slopes in the reach. The practice of the discharge's correction was found to be insignificant when it is applied to reaches with extensive flood protection. Therefore, more cautious considerations must be provided when flood inundation for such reaches produced based on Red LiDAR data.

# References


Apel, H., Martínez Trepat, O., Nghia Hung, N., Thi Chinh, D., Merz, B., & Viet Dung, N. (2016). Combined fluvial and pluvial urban flood hazard analysis: Concept development and application to Can Tho city, Mekong Delta, Vietnam. *Natural Hazards and Earth System Sciences*, *16*(4), 941–961. https://doi.org/10.5194/nhess-16-941-2016

Azizian, A., & Koohi, S. (2021). The effects of applying different DEM resolutions, DEM sources and flow tracing algorithms on LS factor and sediment yield estimation using USLE in Barajin river basin (BRB), Iran. *Paddy and Water Environment*. https://doi.org/10.1007/s10333-021-00847-6

Bhuyian, M. N. M., & Kalyanapu, A. (2018). Accounting digital elevation uncertainty for flood consequence assessment. *Journal of Flood Risk Management*, *11*, S1051–S1062. https://doi.org/10.1111/jfr3.12293

Blöschl, G., Hall, J., Parajka, J., Perdigão, R. A. P., Merz, B., Arheimer, B., Aronica, G. T., Bilibashi, A., Bonacci, O., Borga, M., Ivan, Č., Castellarin, A., & Chirico, G. B. (2017). *European floods*. *590*(August), 588–590.

Bradbrook, K. F., Lane, S. N., Waller, S. G., & Bates, P. D. (2004). Two dimensional diffusion wave modelling of flood inundation using a simplified channel representation. *International Journal of River Basin Management*, *2*(3), 211–223. https://doi.org/10.1080/15715124.2004.9635233

Breili, K., Simpson, M. J. R., Klokkervold, E., & Roaldsdotter Ravndal, O. (2019). High accuracy

coastal flood mapping for Norway using LiDAR data. *Natural Hazards and Earth System Sciences Discussions*, *August*, 1–41. https://doi.org/10.5194/nhess-2019-217
Bures, L., Roub, R., Sychova, P., Gdulova, K., & Doubalova, J. (2019). Comparison of bathymetric data sources used in hydraulic modelling of floods. *Journal of Flood Risk Management*, *12*(S1), 1–10. https://doi.org/10.1111/jfr3.12495
Casas, A., Benito, G., Thorndycraft, V. R., & Rico, M. (2006). The topographic data source of digital terrain models as a key element in the accuracy of hydraulic flood modelling. *Earth Surface Processes and Landforms*, *31*(4), 444–456. https://doi.org/10.1002/esp.1278
Choné, G., Biron, P. M., & Buffin-Bélanger, T. (2018). Flood hazard mapping techniques with LiDAR in the absence of river bathymetry data. *E3S Web of Conferences*, *40*, 1–8. https://doi.org/10.1051/e3sconf/20184006005
Chow, V. T. (1959). *Open Channel Hydraulics*. McGraw-Hill.
Christensen, J. H., & Christensen, O. B. (2003). Severe summertime flooding in Europe. *Nature*, *421*(6925), 805–806. https://doi.org/10.1038/421805a
Ciscar, J. C., Iglesias, A., Feyen, L., Szabó, L., Van Regemorter, D., Amelung, B., Nicholls, R., Watkiss, P., Christensen, O. B., Dankers, R., Garrote, L., Goodess, C. M., Hunt, A., Moreno, A., Richards, J., & Soria, A. (2011). Physical and economic consequences of climate change in Europe. *Proceedings of the National Academy of Sciences of the United States of America*, *108*(7), 2678–2683. https://doi.org/10.1073/pnas.1011612108
Cook, A., & Merwade, V. (2009). Effect of topographic data, geometric configuration and modeling approach on flood inundation mapping. *Journal of Hydrology*, *377*(1–2), 131–142. https://doi.org/10.1016/j.jhydrol.2009.08.015
Delaney, K. B., & Evans, S. G. (2015). The 2000 Yigong landslide (Tibetan Plateau), rockslide-dammed lake and outburst flood: Review, remote sensing analysis, and process modelling. *Geomorphology*, *246*, 377–393. https://doi.org/10.1016/j.geomorph.2015.06.020
Dey, S., Saksena, S., & Merwade, V. (2019). Assessing the effect of different bathymetric models on hydraulic simulation of rivers in data sparse regions. *Journal of Hydrology*, *575*(March), 838–851. https://doi.org/10.1016/j.jhydrol.2019.05.085
Dowman, I. (2004). Integration of LIDAR and IFSAR for mapping. *International Archives of Photogrammetry and Remote Sensing*, *35*(B2), 90–100.
Engeland, K., Glad, P., Hamududu, B. H., Li, H., Reitan, T., & Stenius, S. (2020). *Lokal og regional flomfrekvensanalyse*. https://publikasjoner.nve.no/rapport/2020/rapport2020_10.pdf
Frei, C., Schöll, R., Fukutome, S., Schmidli, J., & Vidale, P. L. (2006). Future change of precipitation extremes in Europe: Intercomparison of scenarios from regional climate models. *Journal of Geophysical Research Atmospheres*, *111*(6). https://doi.org/10.1029/2005JD005965
Hafezi, M., Sahin, O., Stewart, R. A., & Mackey, B. (2018). Creating a novel multi-layered integrative climate change adaptation planning approach using a systematic literature review. *Sustainability (Switzerland)*, *10*(11). https://doi.org/10.3390/su10114100
Hall, J., & Solomatine, D. (2008). A framework for uncertainty analysis in flood risk management decisions. *International Journal of River Basin Management*, *6*(2), 85–98. https://doi.org/10.1080/15715124.2008.9635339
Hilldale, R. C., & Raff, D. (2008). Assessing the ability of airborne LiDAR to map river bathymetry. *EARTH SURFACE PROCESSES AND LANDFORMS*, *33*(5), 773–783. https://doi.org/10.1002/esp.1575
Hodgson, M. E., Jensen, J. R., Schmidt, L., Schill, S., & Davis, B. (2003). An evaluation of LIDAR- and IFSAR-derived digital elevation models in leaf-on conditions with USGS Level 1 and Level 2 DEMs. *Remote Sensing of Environment*, *84*(2), 295–308. https://doi.org/10.1016/S0034-4257(02)00114-1
Irish, J. L., & Lillycrop, W. J. (1999). Scanning laser mapping of the coastal zone: the SHOALS system. *ISPRS JOURNAL OF PHOTOGRAMMETRY AND REMOTE SENSING*, *54*(2–3), 123–129. https://doi.org/10.1016/S0924-2716(99)00003-9
Jakovljević, G., & Govedarica, M. (2019). Water Body Extraction and Flood Risk Assessment Using Lidar and Open Data. In *Climate Change Management*. https://doi.org/10.1007/978-3-030-03383-5_7
Jarihani, A. A., Callow, J. N., McVicar, T. R., Van Niel, T. G., & Larsen, J. R. (2015). Satellite-derived

Digital Elevation Model (DEM) selection, preparation and correction for hydrodynamic modelling in large, low-gradient and data-sparse catchments. *Journal of Hydrology*, *524*, 489–506. https://doi.org/10.1016/j.jhydrol.2015.02.049

Jiang, Y., Zevenbergen, C., & Ma, Y. (2018). Urban pluvial flooding and stormwater management: A contemporary review of China’s challenges and “sponge cities” strategy. *Environmental Science and Policy*, *80*, 132–143. https://doi.org/10.1016/j.envsci.2017.11.016

Jung, H. C., Jasinski, M., Kim, J. W., Shum, C. K., Bates, P., Neal, J., Lee, H., & Alsdorf, D. (2012). Calibration of two-dimensional floodplain modeling in the central Atchafalaya Basin Floodway System using SAR interferometry. *Water Resources Research*, *48*(7), 1–13. https://doi.org/10.1029/2012WR011951

Kinzel, P. J., Legleiter, C. J., & Nelson, J. M. (2013). Mapping River Bathymetry With a Small Footprint Green LiDAR: Applications and Challenges. *JOURNAL OF THE AMERICAN WATER RESOURCES ASSOCIATION*, *49*(1), 183–204. https://doi.org/10.1111/jawr.12008

Kumar, R., & Acharya, P. (2016). Flood hazard and risk assessment of 2014 floods in Kashmir Valley: a space-based multisensor approach. *Natural Hazards*, *84*(1), 437–464. https://doi.org/10.1007/s11069-016-2428-4

Lawrence, D. (2020). Uncertainty introduced by flood frequency analysis in projections for changes in flood magnitudes under a future climate in Norway. *Journal of Hydrology: Regional Studies*, *28*(December 2019), 100675. https://doi.org/10.1016/j.ejrh.2020.100675

Lefsky, M. A., Cohen, W. B., Parker, G. G., & Harding, D. J. (2002). Lidar remote sensing for ecosystem studies. *BIOSCIENCE*, *52*(1), 19–30. https://doi.org/10.1641/0006-3568(2002)052{[}0019:LRSFES]2.0.CO;2

Maksimović, Č., Prodanović, D., Boonya-Aroonnet, S., Leitão, J. P., Djordjević, S., & Allitt, R. (2009). Overland flow and pathway analysis for modelling of urban pluvial flooding. *Journal of Hydraulic Research*, *47*(4), 512–523. https://doi.org/10.1080/00221686.2009.9522027

Mandlburger, G., Hauer, C., Wieser, M., & Pfeifer, N. (2015). Topo-Bathymetric LiDAR for Monitoring River Morphodynamics and Instream Habitats-A Case Study at the Pielach River. *REMOTE SENSING*, *7*(5), 6160–6195. https://doi.org/10.3390/rs70506160

Maruti, S. F., Amerudin, S., Kadir, W. H. W., & Yusof, Z. M. (2018). The validation of hydrodynamic modelling for 2014 flood in Kuala Krai, Kelantan. *MATEC Web of Conferences*, *250*. https://doi.org/10.1051/matecconf/201825004005

McClean, F., Dawson, R., & Kilsby, C. (2020). Implications of Using Global Digital Elevation Models for Flood Risk Analysis in Cities. *Water Resources Research*, *56*(10). https://doi.org/10.1029/2020WR028241

Merz, B., & Thieken, A. H. (2005). Separating natural and epistemic uncertainty in flood frequency analysis. *Journal of Hydrology*, *309*(1–4), 114–132. https://doi.org/10.1016/j.jhydrol.2004.11.015

Muhadi, N. A., Abdullah, A. F., Bejo, S. K., Mahadi, M. R., & Mijic, A. (2020). The use of LiDAR-derived DEM in flood applications: A review. *Remote Sensing*, *12*(14), 1–20. https://doi.org/10.3390/rs12142308

Muthusamy, M., Casado, M. R., Butler, D., & Leinster, P. (2021). Understanding the effects of Digital Elevation Model resolution in urban fluvial flood modelling. *Journal of Hydrology*, *596*(January). https://doi.org/10.1016/j.jhydrol.2021.126088

Reil, A., Skoulikaris, C., Alexandridis, T. K., & Roub, R. (2018). Evaluation of riverbed representation methods for one-dimensional flood hydraulics model. *JOURNAL OF FLOOD RISK MANAGEMENT*, *11*(2), 169–179. https://doi.org/10.1111/jfr3.12304

Rözer, V., Müller, M., Bubeck, P., Kienzler, S., Thieken, A., Pech, I., Schröter, K., Buchholz, O., & Kreibich, H. (2016). Coping with pluvial floods by private households. *Water (Switzerland)*, *8*(7). https://doi.org/10.3390/W8070304

Tanaka, T., Kiyohara, K., & Tachikawa, Y. (2020). Comparison of fluvial and pluvial flood risk curves in urban cities derived from a large ensemble climate simulation dataset: A case study in Nagoya, Japan. *Journal of Hydrology*, *584*(September 2019), 124706. https://doi.org/10.1016/j.jhydrol.2020.124706

The US Army Corps of Engineers, T. H. E. C. (2021). *HEC-RAS 6.0*. https://www.hec.usace.army.mil/software/hec-ras/

Thomas Steven Savage, J., Pianosi, F., Bates, P., Freer, J., & Wagener, T. (2016). Quantifying the importance of spatial resolution and other factors through global sensitivity analysis of a flood inundation model. *Water Resources Research*, *52*(11), 9146–9163. https://doi.org/10.1002/2015WR018198

Wang, Y., Chen, A. S., Fu, G., Djordjević, S., Zhang, C., & Savić, D. A. (2018). An integrated framework for high-resolution urban flood modelling considering multiple information sources and urban features. *Environmental Modelling and Software*, *107*(July 2017), 85–95. https://doi.org/10.1016/j.envsoft.2018.06.010

Zhang, Y. (2020). Using LiDAR-DEM based rapid flood inundation modelling framework to map floodplain inundation extent and depth. *Journal of Geographical Sciences*, *30*(10), 1649–1663. https://doi.org/10.1007/s11442-020-1805-9

Zhou, Q., Mikkelsen, P. S., Halsnæs, K., & Arnbjerg-Nielsen, K. (2012). Framework for economic pluvial flood risk assessment considering climate change effects and adaptation benefits. *Journal of Hydrology*, *414–415*, 539–549. https://doi.org/10.1016/j.jhydrol.2011.11.031

Zurich. (2020). *Three common types of flood explained*. https://www.zurich.com/en/knowledge/topics/flood-and-water-damage/three-common-types-of-flood

# Appendix

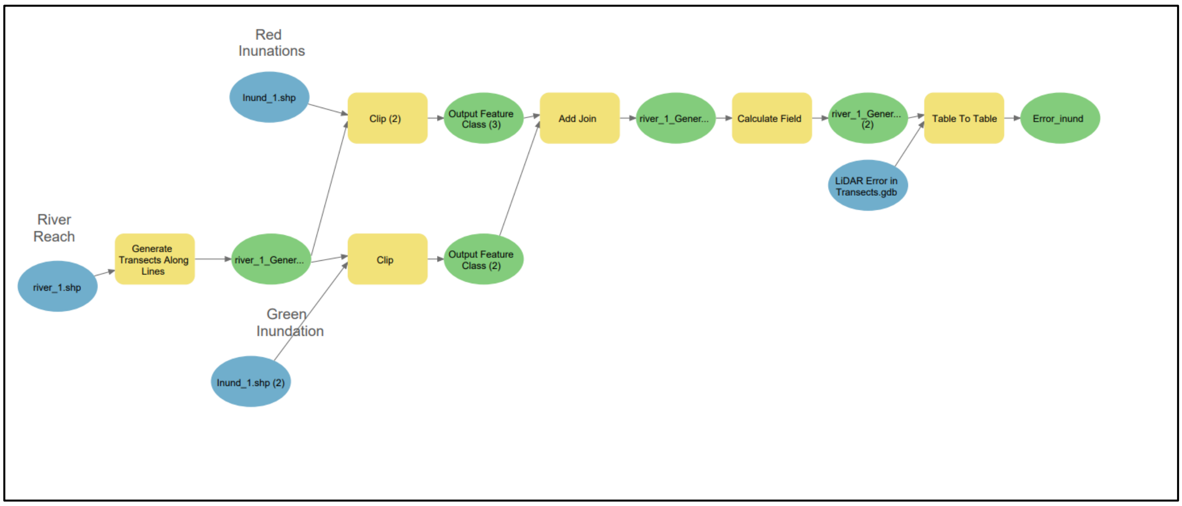
## Appendix A: Methodology's developments

Background:

The final developed methodology in the manuscript has been the output of a series of trails, especially for the bank's slope estimations. Each approach includes ArcGIS ModelBuilder, Python code for automation, and R language codes for postprocessing the data. In this section, brief enlightens of the involved steps will be shown.

LiDAR Error in transects

1. ModelBuilder chart




2. Python code for the automation

```
# -*- coding: utf-8 -*-
"""
Generated by ArcGIS ModelBuilder on : 2021-03-02 19:01:19
"""
import arcpy
var1 = [1,2,3,4, 5, 6, 7, 8, 9]
var2 = [1,2, 3, 4, 5, 6, 7]
var3 = [850, 550, 2000, 1500, 1500, 800, 500, 500, 750]  # The Suitable
length of Transect in meters

    # To allow overwriting outputs change overwriteOutput option to True.
arcpy.env.overwriteOutput = True

for i in var1:

    arcpy.ImportToolbox(r"c:\program
files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management
Tools.tbx")
    river_1_shp = r"E:\GIS Automation\River Lines Backup" + r"\river_" +
str(i) + r".shp"
    for j in var2:
```

```
        Inund_1_shp_2_ = "E:\\GIS Automation\\LiDAR Error
Transects\\Site_" + str(i) + "\\green\\Inund_" + str(j) + ".shp"
        Inund_1_shp = "E:\\GIS Automation\\LiDAR Error Transects\\Site_"
+ str(i) + "\\red\\Inund_" + str(j) + ".shp"
        LiDAR_Error_in_Transects_gdb = r"E:\GIS Automation\LiDAR Error
Transects\Site_" + str(i) + r"\output"

        # Process: Generate Transects Along Lines (Generate Transects
Along Lines) (management)
        river_1_GenerateTransectsAlongLines1_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\LiDAR Error in
Transects\\river_1_GenerateTransectsAlongLines1.shp"

arcpy.management.GenerateTransectsAlongLines(in_features=river_1_shp,
out_feature_class=river_1_GenerateTransectsAlongLines1_shp, interval="50
Meters",  transect_length="{} Meters".format(var3[i-1]),
include_ends="NO_END_POINTS")

        # Process: Clip (Clip) (analysis)
        Output_Feature_Class_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\LiDAR Error in
Transects\\LiDAR Error in Transects.gdb\\river_1_GenerateTransectsAlo"

arcpy.analysis.Clip(in_features=river_1_GenerateTransectsAlongLines1_shp,
clip_features=Inund_1_shp_2_, out_feature_class=Output_Feature_Class_2_,
cluster_tolerance="")

        # Process: Clip (2) (Clip) (analysis)
        Output_Feature_Class_3_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\LiDAR Error in
Transects\\LiDAR Error in Transects.gdb\\river_1_GenerateTransectsAlo1"

arcpy.analysis.Clip(in_features=river_1_GenerateTransectsAlongLines1_shp,
clip_features=Inund_1_shp, out_feature_class=Output_Feature_Class_3_,
cluster_tolerance="")

        # Process: Add Join (Add Join) (management)
        river_1_GenerateTransectsAlo =
arcpy.management.AddJoin(in_layer_or_view=Output_Feature_Class_2_,
in_field="OBJECTID", join_table=Output_Feature_Class_3_,
join_field="OBJECTID", join_type="KEEP_ALL")[0]

        # Process: Calculate Field (Calculate Field) (management)
        river_1_GenerateTransectsAlo_2_ =
arcpy.management.CalculateField(in_table=river_1_GenerateTransectsAlo,
field="Error_Inund",
expression="100*(!river_1_GenerateTransectsAlo1.Shape_Length!-
!river_1_GenerateTransectsAlo.Shape_Length!)/(!river_1_GenerateTransectsA
lo.Shape_Length!)", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]
        FFF = "LiDAR Error Transect_ClipDS_" + str(j) + ".csv"
        # Process: Table To Table (Table To Table) (conversion)
        Error_inund =
arcpy.conversion.TableToTable(in_rows=river_1_GenerateTransectsAlo_2_,
out_path=LiDAR_Error_in_Transects_gdb, out_name= FFF, where_clause="",
field_mapping="Id \"Id\" true true false 4 Long 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo.Id,-
1,-1;ORIG_FID \"ORIG_FID\" true true false 4 Long 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo.ORIG_
FID,-1,-1;Shape_Length \"Shape_Length\" false true true 8 Double 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo.Shape
```

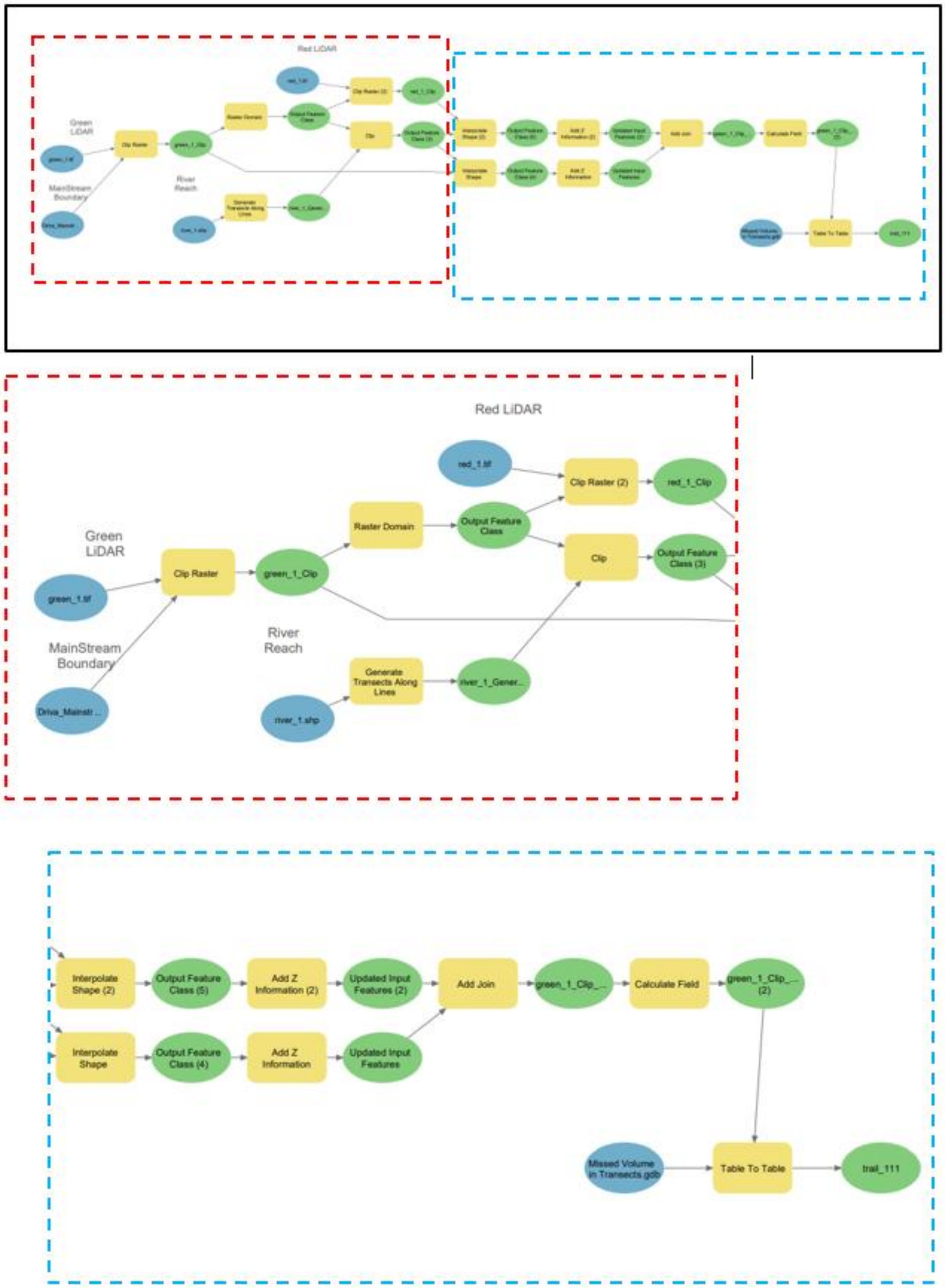
```
_Length,-1,-1;Error_Inund \"Error_Inund\" true true false 512 Text 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo.Error
_Inund,0,512;OBJECTID \"OBJECTID\" false true false 4 Long 0
9,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo1.OBJE
CTID,-1,-1;Id_1 \"Id\" true true false 4 Long 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo1.Id,-
1,-1;ORIG_FID_1 \"ORIG_FID\" true true false 4 Long 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo1.ORIG
_FID,-1,-1;Shape_Length_1 \"Shape_Length\" false true true 8 Double 0
0,First,#,river_1_GenerateTransectsAlo,river_1_GenerateTransectsAlo1.Shap
e_Length,-1,-1", config_keyword="")[0]
    print("finished " + str(i) + " ^_^")
```

3. R code for the postprocessing

```
dir <- "E://GIS Automation//LiDAR Error Transects//Site_"
x <- "//output//LiDAR Error Transect_ClipDS_"
y <- ".csv"
for (i in 1:1){
  #data_all <- NULL

  for (j in 1:7){
    data_dir <- paste(dir,i,x,j,y, sep = "")
    output <- paste(dir, i, x,"T_",i,".csv", sep = "")
    m <- read.csv(data_dir,header = T, sep = ",")
    print(dim(m))
    if(j==1){
      data_all= cbind(m$OBJECTID,m$Error_Inund)
    }
    else {
      data_all = cbind(data_all, m$Error_Inund)
    }
    #data_all <- cbind(m, data_all)
    print(j)

  }
  write.table(data_all,output, sep = ",", col.names = T, row.names = F)
  print("finished")
}
View(m)
```

Missed LiDAR Volume

1. ModelBuilder chart



2. Python code for the automation

```
3. # -*- coding: utf-8 -*-
   """
   Generated by ArcGIS ModelBuilder on : 2021-03-02 15:53:28
   """
   import arcpy

   #def Model():  # Model
   var1 = [10,11]
   var2 = [2, 3, 4, 5, 6, 7]
   # To allow overwriting outputs change overwriteOutput option to
   True.
   arcpy.env.overwriteOutput = True


   for i in var1:
       arcpy.ImportToolbox(r"c:\program
   files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management
   Tools.tbx")
       green_1_tif = arcpy.Raster("E:\\GIS Automation\\Full Terrain
   Models\\Terrain_" + str(i) + ".tif")
       Driva_Mainstream_shp = r"E:\GIS Automation\Missed Volume
   Transects\Sites\mainstream boundary\boundary_" + str(i) + r".shp"
       river_1_shp = r"E:\GIS Automation\River Lines Backup" +
   r"\river_" + str(i) + r".shp"
       red_1_tif = arcpy.Raster("E:\\GIS Automation\\Bank
   Slope_5\\Site_" + str(i) + "\\RED LIDAR\\RED_LIDAR.tif")
       Missed_Volume_in_Transects_gdb = r"E:\GIS Automation\Missed
   Volume Transects\Sites\output"

       # Process: Clip Raster (Clip Raster) (management)
       green_1_Clip =
   "C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
   Transects\\Missed Volume in Transects.gdb\\green_1_Clip"
       arcpy.management.Clip(in_raster=green_1_tif,
   rectangle="477199.898508832 6946208.66544873 485612.001650867
   6949328.81936749", out_raster=green_1_Clip,
   in_template_dataset=Driva_Mainstream_shp, nodata_value="-
   3.402823e+38", clipping_geometry="ClippingGeometry",
   maintain_clipping_extent="NO_MAINTAIN_EXTENT")
       green_1_Clip = arcpy.Raster(green_1_Clip)

       # Process: Generate Transects Along Lines (Generate Transects
   Along Lines) (management)
       river_1_GenerateTransectsAlongLines1_shp =
   "C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
   Transects\\river_1_GenerateTransectsAlongLines3.shp"

   arcpy.management.GenerateTransectsAlongLines(in_features=river_1_sh
   p, out_feature_class=river_1_GenerateTransectsAlongLines1_shp,
   interval="50 Meters", transect_length="200 Meters",
   include_ends="NO_END_POINTS")

       # Process: Raster Domain (Raster Domain) (3d)
       Output_Feature_Class =
   "C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
   Transects\\Missed Volume in
```

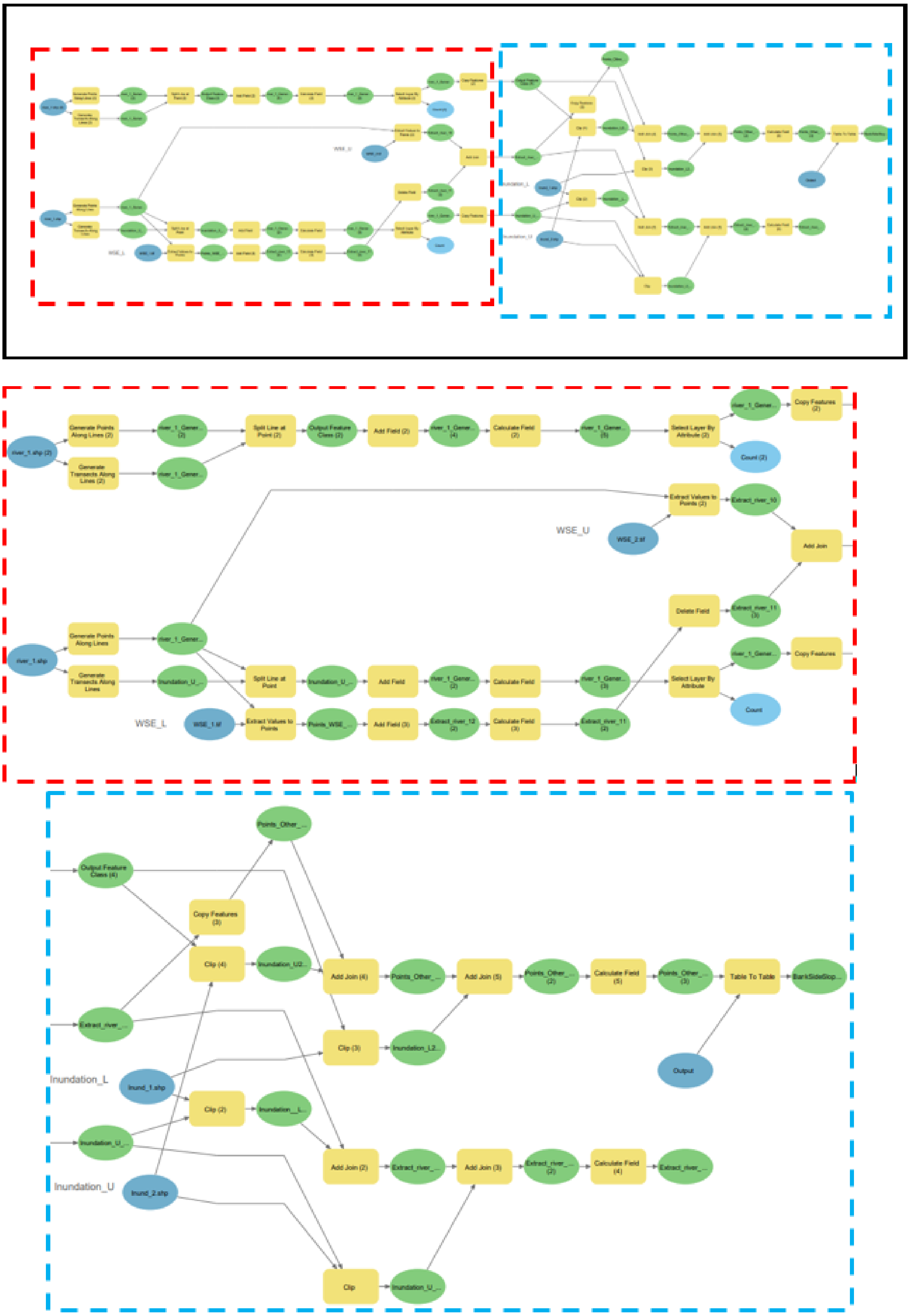
```
Transects.gdb\\green_1_Clip_RasterDomain"
    arcpy.ddd.RasterDomain(in_raster=green_1_Clip,
out_feature_class=Output_Feature_Class,
out_geometry_type="POLYGON")

    # Process: Clip (Clip) (analysis)
    Output_Feature_Class_3_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
Transects\\Missed Volume in
Transects.gdb\\river_1_GenerateTransectsAlo"

arcpy.analysis.Clip(in_features=river_1_GenerateTransectsAlongLines
1_shp, clip_features=Output_Feature_Class,
out_feature_class=Output_Feature_Class_3_, cluster_tolerance="")

    # Process: Interpolate Shape (Interpolate Shape) (3d)
    Output_Feature_Class_4_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
Transects\\Missed Volume in
Transects.gdb\\green_1_Clip_InterpolateShap"
    arcpy.ddd.InterpolateShape(in_surface=green_1_Clip,
in_feature_class=Output_Feature_Class_3_,
out_feature_class=Output_Feature_Class_4_, sample_distance=None,
z_factor=1, method="BILINEAR", vertices_only="DENSIFY",
pyramid_level_resolution=0, preserve_features="EXCLUDE")

    # Process: Add Z Information (Add Z Information) (3d)
    Updated_Input_Features =
arcpy.ddd.AddZInformation(in_feature_class=Output_Feature_Class_4_,
out_property=["Z_MEAN"], noise_filtering="")[0]

    # Process: Clip Raster (2) (Clip Raster) (management)
    red_1_Clip =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
Transects\\Missed Volume in Transects.gdb\\red_1_Clip"
    arcpy.management.Clip(in_raster=red_1_tif,
rectangle="477199.9827 6946364.8659 485354.732700001 6949328.6159",
out_raster=red_1_Clip, in_template_dataset=Output_Feature_Class,
nodata_value="-3.402823e+38", clipping_geometry="ClippingGeometry",
maintain_clipping_extent="NO_MAINTAIN_EXTENT")
    red_1_Clip = arcpy.Raster(red_1_Clip)

    # Process: Interpolate Shape (2) (Interpolate Shape) (3d)
    Output_Feature_Class_5_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Missed Volume in
Transects\\Missed Volume in
Transects.gdb\\red_1_Clip_InterpolateShape"
    arcpy.ddd.InterpolateShape(in_surface=red_1_Clip,
in_feature_class=Output_Feature_Class_3_,
out_feature_class=Output_Feature_Class_5_, sample_distance=None,
z_factor=1, method="BILINEAR", vertices_only="DENSIFY",
pyramid_level_resolution=0, preserve_features="EXCLUDE")

    # Process: Add Z Information (2) (Add Z Information) (3d)
    Updated_Input_Features_2_ =
arcpy.ddd.AddZInformation(in_feature_class=Output_Feature_Class_5_,
out_property=["Z_MEAN"], noise_filtering="")[0]

    # Process: Add Join (Add Join) (management)
    green_1_Clip_InterpolateShap =
arcpy.management.AddJoin(in_layer_or_view=Updated_Input_Features,
```

```
in_field="OBJECTID", join_table=Updated_Input_Features_2_,
join_field="OBJECTID", join_type="KEEP_ALL")[0]

    # Process: Calculate Field (Calculate Field) (management)
    green_1_Clip_InterpolateShap_2_ =
arcpy.management.CalculateField(in_table=green_1_Clip_InterpolateSh
ap, field="Vol_Diff",
expression="!red_1_Clip_InterpolateShape.Z_Mean!-
!green_1_Clip_InterpolateShap.Z_Mean!", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

    # Process: Table To Table (Table To Table) (conversion)
    output_table = "Missed Volume Transect_" + str(i) + ".csv"
    trail_111 =
arcpy.conversion.TableToTable(in_rows=green_1_Clip_InterpolateShap_
2_, out_path=Missed_Volume_in_Transects_gdb, out_name=
output_table, where_clause="", field_mapping="Id \"Id\" true true
false 4 Long 0
0,First,#,green_1_Clip_InterpolateShap2,green_1_Clip_InterpolateSha
p.Id,-1,-1;ORIG_FID \"ORIG_FID\" true true false 4 Long 0
0,First,#,green_1_Clip_InterpolateShap2,green_1_Clip_InterpolateSha
p.ORIG_FID,-1,-1;Shape_Length \"Shape_Length\" false true true 8
Double 0
0,First,#,green_1_Clip_InterpolateShap2,green_1_Clip_InterpolateSha
p.Shape_Length,-1,-1;Z_Mean \"Z_Mean\" true true false 8 Double 0
0,First,#,green_1_Clip_InterpolateShap2,green_1_Clip_InterpolateSha
p.Z_Mean,-1,-1;Vol_Diff \"Vol_Diff\" true true false 512 Text 0
0,First,#,green_1_Clip_InterpolateShap2,green_1_Clip_InterpolateSha
p.Vol_Diff,0,512;OBJECTID \"OBJECTID\" false true false 4 Long 0
9,First,#,green_1_Clip_InterpolateShap2,red_1_Clip_InterpolateShape
.OBJECTID,-1,-1;Id_1 \"Id\" true true false 4 Long 0
0,First,#,green_1_Clip_InterpolateShap2,red_1_Clip_InterpolateShape
.Id,-1,-1;ORIG_FID_1 \"ORIG_FID\" true true false 4 Long 0
0,First,#,green_1_Clip_InterpolateShap2,red_1_Clip_InterpolateShape
.ORIG_FID,-1,-1;Shape_Length_1 \"Shape_Length\" false true true 8
Double 0
0,First,#,green_1_Clip_InterpolateShap2,red_1_Clip_InterpolateShape
.Shape_Length,-1,-1;Z_Mean_1 \"Z_Mean\" true true false 8 Double 0
0,First,#,green_1_Clip_InterpolateShap2,red_1_Clip_InterpolateShape
.Z_Mean,-1,-1", config_keyword="")[0]
    print("finished " + str(i) + "^_^")
```

Bank’s slope approaches using *__Transects__*

Approach 1

1. ModelBuilder chart



2. Python code for the automation

```
# -*- coding: utf-8 -*-
"""
Generated by ArcGIS ModelBuilder on : 2021-03-01 19:06:56
"""
import arcpy

#def Model():  # Model

var1 = [6,7]
var2 = [2, 3, 4, 5, 6, 7,8,9,10]
# To allow overwriting outputs change overwriteOutput option to True.
arcpy.env.overwriteOutput = True

# Check out any necessary licenses.arcpy.CheckOutExtension("spatial")

arcpy.ImportToolbox(r"c:\program
files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management
Tools.tbx")

for i in var1:
    river_1_shp = r"E:\GIS Automation\Bank Slope_2\Site_" + str(i) +
r"\river_" + str(i) + r".shp"
    river_1_shp_2_ = r"E:\GIS Automation\Bank Slope_2\Site_" + str(i) +
r"\river_" + str(i) + r".shp"
    for j in var2:
        m = j - 1
        n = j
        WSE_L_Path = r"E:\GIS Automation\Bank Slope_3\Site_" + str(i) +
r"\IDW_M\Idw_M" + str(m) + r"_boundary.tif"
        WSE_U_Path = r"E:\GIS Automation\Bank Slope_3\Site_" + str(i) +
r"\IDW_M\Idw_M" + str(n) + r"_boundary.tif"
        Inund_L_Path = r"E:\GIS Automation\Bank Slope_3\Site_" + str(i) +
r"\Domain\Idw_M" + str(m) + r"_domain.shp"
        Inund_U_Path = r"E:\GIS Automation\Bank Slope_3\Site_" + str(i) +
r"\Domain\Idw_M" + str(n) + r"_domain.shp"

        WSE_2_tif = arcpy.Raster(WSE_U_Path)
        WSE_1_tif = arcpy.Raster(WSE_L_Path)
        Inund_1_shp = Inund_L_Path
        Inund_2_shp = Inund_U_Path
        Bank_Slope_Second_Approach_gdb =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb"
        Output = r"E:\GIS Automation\Bank Slope_3\Site_" + str(i) +
r"\Output"

        # Process: Generate Transects Along Lines (Generate Transects
Along Lines) (management)
        Inundation_U_T_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Inundation_U_T.shp"

arcpy.management.GenerateTransectsAlongLines(in_features=river_1_shp,
```

```
out_feature_class=Inundation_U_T_shp,
                                                                     interval="50
Meters", transect_length="2000 Meters",

include_ends="NO_END_POINTS")

        # Process: Generate Points Along Lines (Generate Points Along
Lines) (management)
        river_1_GeneratePointsAlongLines1_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\river_1_GeneratePointsAlongLines19.shp"

arcpy.management.GeneratePointsAlongLines(Input_Features=river_1_shp,

Output_Feature_Class=river_1_GeneratePointsAlongLines1_shp,

Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
                                                                Include_End_Points="")

        # Process: Split Line at Point (Split Line at Point) (management)
        Inundation_U_T_Splitted =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\Inundation_U_T_Splitted"
        arcpy.management.SplitLineAtPoint(in_features=Inundation_U_T_shp,

point_features=river_1_GeneratePointsAlongLines1_shp,

out_feature_class=Inundation_U_T_Splitted, search_radius="")

        # Process: Add Field (Add Field) (management)
        river_1_GenerateTransectsAlo_2_ = \
        arcpy.management.AddField(in_table=Inundation_U_T_Splitted,
field_name="Cop_ID", field_type="LONG",
                                  field_precision=None, field_scale=None,
field_length=None, field_alias="",
                                  field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Calculate Field (Calculate Field) (management)
        river_1_GenerateTransectsAlo_3_ = \

arcpy.management.CalculateField(in_table=river_1_GenerateTransectsAlo_2_,
field="Cop_ID",
                                        expression="!OBJECTID!",
expression_type="PYTHON3", code_block="",
                                        field_type="TEXT")[0]

        # Process: Select Layer By Attribute (Select Layer By Attribute)
(management)
        river_1_GenerateTransectsAlo5, Count =
arcpy.management.SelectLayerByAttribute(
            in_layer_or_view=river_1_GenerateTransectsAlo_3_,
selection_type="NEW_SELECTION",
            where_clause="MOD(OBJECTID,2) = 0", invert_where_clause="")

        # Process: Generate Transects Along Lines (2) (Generate Transects
Along Lines) (management)
        river_1_GenerateTransectsAlongLines18_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\river_1_GenerateTransectsAlongLines18.shp"
```

```
arcpy.management.GenerateTransectsAlongLines(in_features=river_1_shp_2_,

out_feature_class=river_1_GenerateTransectsAlongLines18_shp,
                                                      interval="50
Meters", transect_length="2000 Meters",

include_ends="NO_END_POINTS")

        # Process: Generate Points Along Lines (2) (Generate Points Along
Lines) (management)
        river_1_GeneratePointsAlongLines1_shp_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\river_1_GeneratePointsAlongLines4.shp"

arcpy.management.GeneratePointsAlongLines(Input_Features=river_1_shp_2_,

Output_Feature_Class=river_1_GeneratePointsAlongLines1_shp_2_,

Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
                                                   Include_End_Points="")

        # Process: Split Line at Point (2) (Split Line at Point)
(management)
        Output_Feature_Class_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\river_1_GenerateTransectsAlo"

arcpy.management.SplitLineAtPoint(in_features=river_1_GenerateTransectsAl
ongLines18_shp,

point_features=river_1_GeneratePointsAlongLines1_shp_2_,

out_feature_class=Output_Feature_Class_2_, search_radius="")

        # Process: Add Field (2) (Add Field) (management)
        river_1_GenerateTransectsAlo_4_ = \
        arcpy.management.AddField(in_table=Output_Feature_Class_2_,
field_name="Cop_ID", field_type="LONG",
                                  field_precision=None, field_scale=None,
field_length=None, field_alias="",
                                  field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Calculate Field (2) (Calculate Field) (management)
        river_1_GenerateTransectsAlo_5_ = \

arcpy.management.CalculateField(in_table=river_1_GenerateTransectsAlo_4_,
field="Cop_ID",
                                        expression="!OBJECTID!",
expression_type="PYTHON3", code_block="",
                                        field_type="TEXT")[0]

        # Process: Select Layer By Attribute (2) (Select Layer By
Attribute) (management)
        river_1_GenerateTransectsAlo7, Count_2_ =
arcpy.management.SelectLayerByAttribute(
            in_layer_or_view=river_1_GenerateTransectsAlo_5_,
selection_type="NEW_SELECTION",
            where_clause="MOD(OBJECTID,2)<>0", invert_where_clause="")

        # Process: Extract Values to Points (2) (Extract Values to
```

```
Points) (sa)
        Extract_river_10 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\Extract_river_10"

arcpy.sa.ExtractValuesToPoints(in_point_features=river_1_GeneratePointsAl
ongLines1_shp, in_raster=WSE_2_tif,

out_point_features=Extract_river_10, interpolate_values="NONE",
                                       add_attributes="VALUE_ONLY")

        # Process: Extract Values to Points (Extract Values to Points)
(sa)
        Points_WSE_Low =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\Points_WSE_Low"

arcpy.sa.ExtractValuesToPoints(in_point_features=river_1_GeneratePointsAl
ongLines1_shp, in_raster=WSE_1_tif,
                                       out_point_features=Points_WSE_Low,
interpolate_values="NONE",
                                       add_attributes="VALUE_ONLY")

        # Process: Add Field (3) (Add Field) (management)
        Extract_river_12_2_ = \
        arcpy.management.AddField(in_table=Points_WSE_Low,
field_name="Low_Elev", field_type="DOUBLE",
                                  field_precision=None, field_scale=None,
field_length=None, field_alias="",
                                  field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Calculate Field (3) (Calculate Field) (management)
        Extract_river_11_2_ = \
        arcpy.management.CalculateField(in_table=Extract_river_12_2_,
field="Low_Elev", expression="!RASTERVALU!",
                                        expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

        # Process: Delete Field (Delete Field) (management)
        Extract_river_11_3_ =
arcpy.management.DeleteField(in_table=Extract_river_11_2_,
drop_field=["RASTERVALU"])[0]

        # Process: Add Join (Add Join) (management)
        Extract_river_10_Layer = \
        arcpy.management.AddJoin(in_layer_or_view=Extract_river_10,
in_field="OBJECTID", join_table=Extract_river_11_3_,
                                 join_field="OBJECTID",
join_type="KEEP_ALL")[0]

        # Process: Copy Features (Copy Features) (management)
        Inundation_U_T_Splitted_CopyFeatures =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second
Approach.gdb\\Inundation_U_T_Splitted_CopyFeatures"

arcpy.management.CopyFeatures(in_features=river_1_GenerateTransectsAlo5,

out_feature_class=Inundation_U_T_Splitted_CopyFeatures,
config_keyword="",
```

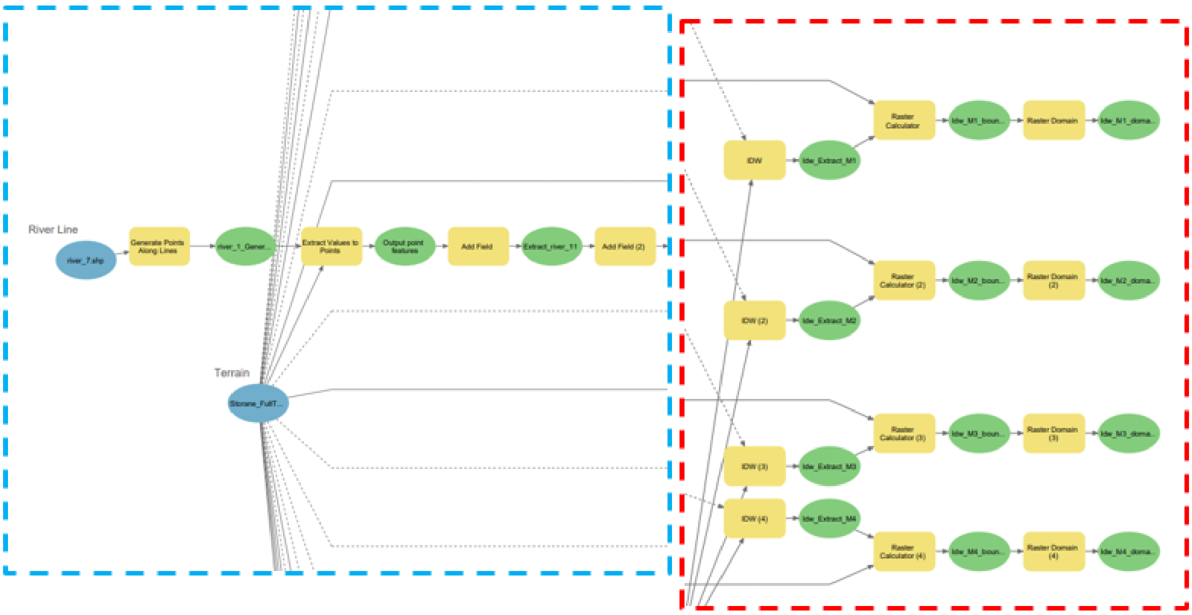
```
                                      spatial_grid_1=None,
spatial_grid_2=None, spatial_grid_3=None)

        # Process: Clip (2) (Clip) (analysis)
        Inundation_L_clipped =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\Inundation__L_clipped"

arcpy.analysis.Clip(in_features=Inundation_U_T_Splitted_CopyFeatures,
clip_features=Inund_1_shp,
                            out_feature_class=Inundation_L_clipped,
cluster_tolerance="")

        # Process: Add Join (2) (Add Join) (management)
        Extract_river_10_Layer1 = \
        arcpy.management.AddJoin(in_layer_or_view=Extract_river_10_Layer,
in_field="Extract_river_10.OBJECTID",
                                 join_table=Inundation_L_clipped,
join_field="OBJECTID", join_type="KEEP_ALL")[0]

        # Process: Clip (Clip) (analysis)
        Inundation_U_clipped =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\Inundation_U_clipped"

arcpy.analysis.Clip(in_features=Inundation_U_T_Splitted_CopyFeatures,
clip_features=Inund_2_shp,
                            out_feature_class=Inundation_U_clipped,
cluster_tolerance="")

        # Process: Add Join (3) (Add Join) (management)
        Extract_river_10_Layer1_2_ = \

arcpy.management.AddJoin(in_layer_or_view=Extract_river_10_Layer1,
in_field="Extract_river_10.OBJECTID",
                                 join_table=Inundation_U_clipped,
join_field="OBJECTID", join_type="KEEP_ALL")[0]

        # Process: Calculate Field (4) (Calculate Field) (management)
        Extract_river_10_Layer7 =
arcpy.management.CalculateField(in_table=Extract_river_10_Layer1_2_,
field="Z1",

expression="(!Inundation_U_clipped.Shape_Length!-
!Inundation__L_clipped.Shape_Length!)/(!Extract_river_10.RASTERVALU!-
!Points_WSE_Low.Low_Elev!)",

expression_type="PYTHON3", code_block="",

field_type="TEXT")[0]

        # Process: Copy Features (3) (Copy Features) (management)
        Points_Other_Side =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second
Approach\\Bank Slope Second Approach.gdb\\Points_Other_Side"
        arcpy.management.CopyFeatures(in_features=Extract_river_10_Layer,
out_feature_class=Points_Other_Side,
                                      config_keyword="",
spatial_grid_1=None, spatial_grid_2=None, spatial_grid_3=None)

        # Process: Copy Features (2) (Copy Features) (management)
```

```
        Output_Feature_Class_4_ = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second 
Approach\\Bank Slope Second 
Approach.gdb\\river_1_GenerateTransectsAlo_CopyFeatures1"

arcpy.management.CopyFeatures(in_features=river_1_GenerateTransectsAlo7, 

out_feature_class=Output_Feature_Class_4_, config_keyword="", 
spatial_grid_1=None,
                                      spatial_grid_2=None, 
spatial_grid_3=None)

        # Process: Clip (4) (Clip) (analysis)
        Inundation_U2_clipped = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second 
Approach\\Bank Slope Second Approach.gdb\\Inundation_U2_clipped"
        arcpy.analysis.Clip(in_features=Output_Feature_Class_4_, 
clip_features=Inund_2_shp,
                            out_feature_class=Inundation_U2_clipped, 
cluster_tolerance="")

        # Process: Add Join (4) (Add Join) (management)
        Points_Other_Side_Layer1 = 
arcpy.management.AddJoin(in_layer_or_view=Points_Other_Side, 
in_field="OBJECTID",

join_table=Inundation_U2_clipped, join_field="OBJECTID",

join_type="KEEP_ALL")[0]

        # Process: Clip (3) (Clip) (analysis)
        Inundation_L2_clipped = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Second 
Approach\\Bank Slope Second Approach.gdb\\Inundation_L2_clipped"
        arcpy.analysis.Clip(in_features=Output_Feature_Class_4_, 
clip_features=Inund_1_shp,
                            out_feature_class=Inundation_L2_clipped, 
cluster_tolerance="")

        # Process: Add Join (5) (Add Join) (management)
        Points_Other_Side_Layer1_2_ = \

arcpy.management.AddJoin(in_layer_or_view=Points_Other_Side_Layer1, 
in_field="Points_Other_Side.OBJECTID",
                                 join_table=Inundation_L2_clipped, 
join_field="OBJECTID", join_type="KEEP_ALL")[0]

        # Process: Calculate Field (5) (Calculate Field) (management)
        Points_Other_Side_Layer1_3_ = 
arcpy.management.CalculateField(in_table=Points_Other_Side_Layer1_2_, 
field="Z2",

expression="(!Inundation_U2_clipped.Shape_Length!-
!Inundation_L2_clipped.Shape_Length!)/(!Points_Other_Side.Extract_river_1
0_RASTERVALU!-!Points_Other_Side.Points_WSE_Low_Low_Elev!)",

expression_type="PYTHON3", code_block="",

field_type="TEXT")[0]

        # Process: Table To Table (Table To Table) (conversion)
```

```
        FFF = "BankSideSlopes3_" + str(i) + str(m) + ".csv"
        BankSideSlopes_csv = \

arcpy.conversion.TableToTable(in_rows=Points_Other_Side_Layer1_3_,
out_path=Output, out_name=FFF,
                                      where_clause="",
                                      field_mapping="OBJECTID
\"OBJECTID\" false true false 4 Long 0
9,First,#,Points_Other_Side_Layer2,Inundation_U2_clipped.OBJECTID,-1,-
1;Extract_river_10_Z1 \"Z1\" true true false 512 Text 0
0,First,#,Points_Other_Side_Layer2,Points_Other_Side.Extract_river_10_Z1,
0,512;Z2 \"Z2\" true true false 512 Text 0
0,First,#,Points_Other_Side_Layer2,Points_Other_Side.Z2,0,512",
                                      config_keyword="")[0]
    print("You Finished " + str(i) + " Now ^^" )
```

Approach 2

1. ModelBuilder chart



2. Python code for the automation
    a. For the 10 layers 'creation

```
# -*- coding: utf-8 -*-
"""
Generated by ArcGIS ModelBuilder on : 2021-03-08 07:29:13
"""
import arcpy
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *

def Model():  # Model

    # To allow overwriting outputs change overwriteOutput option to True.
    arcpy.env.overwriteOutput = False

    # Check out any necessary licenses.
    arcpy.CheckOutExtension("spatial")
    arcpy.CheckOutExtension("3D")
    arcpy.CheckOutExtension("ImageAnalyst")

    arcpy.ImportToolbox(r"c:\program
files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management
Tools.tbx")
    # Model Environment settings
    with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
        river_3_shp = "E:\\GIS Automation\\Bank
Slope_2\\Site_3\\river_3.shp"
        Gaulfoss_FullTerrain_tif = arcpy.Raster("E:\\LiDAR River
Volume\\Gaulfoss\\Final Results\\Merged
Terrain\\Gaulfoss_FullTerrain.tif")

        # Process: Generate Points Along Lines (Generate Points Along
Lines) (management)
        river_1_GeneratePointsAlongLines1_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\river_3_GeneratePointsAlongLines1.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):

arcpy.management.GeneratePointsAlongLines(Input_Features=river_3_shp,
Output_Feature_Class=river_1_GeneratePointsAlongLines1_shp,
Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
Include_End_Points="")
```

```
        # Process: Extract Values to Points (Extract Values to Points)
(sa)
        Output_point_features =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Extract_river_31"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):

arcpy.sa.ExtractValuesToPoints(in_point_features=river_1_GeneratePointsAl
ongLines1_shp, in_raster=Gaulfoss_FullTerrain_tif,
out_point_features=Output_point_features, interpolate_values="NONE",
add_attributes="VALUE_ONLY")

        # Process: Add Field (Add Field) (management)
        Extract_river_11 =
arcpy.management.AddField(in_table=Output_point_features,
field_name="M1", field_type="DOUBLE", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

        # Process: Add Field (2) (Add Field) (management)
        Extract_river_11_2_ =
arcpy.management.AddField(in_table=Extract_river_11, field_name="M2",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (3) (Add Field) (management)
        Extract_river_11_3_ =
arcpy.management.AddField(in_table=Extract_river_11_2_, field_name="M3",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (4) (Add Field) (management)
        Extract_river_11_4_ =
arcpy.management.AddField(in_table=Extract_river_11_3_, field_name="M4",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (5) (Add Field) (management)
        Extract_river_11_5_ =
arcpy.management.AddField(in_table=Extract_river_11_4_, field_name="M5",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (6) (Add Field) (management)
        Extract_river_11_6_ =
arcpy.management.AddField(in_table=Extract_river_11_5_, field_name="M6",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (7) (Add Field) (management)
        Extract_river_11_7_ =
```

```
arcpy.management.AddField(in_table=Extract_river_11_6_, field_name="M7",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (8) (Add Field) (management)
        Extract_river_11_8_ =
arcpy.management.AddField(in_table=Extract_river_11_7_, field_name="M8",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (9) (Add Field) (management)
        Extract_river_11_9_ =
arcpy.management.AddField(in_table=Extract_river_11_8_, field_name="M9",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Add Field (10) (Add Field) (management)
        Extract_river_11_10_ =
arcpy.management.AddField(in_table=Extract_river_11_9_, field_name="M10",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

        # Process: Calculate Field (Calculate Field) (management)
        Extract_river_21 =
arcpy.management.CalculateField(in_table=Extract_river_11_10_,
field="M1", expression="!RASTERVALU!+0.2", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

        # Process: Calculate Field (2) (Calculate Field) (management)
        Extract_river_21_2_ =
arcpy.management.CalculateField(in_table=Extract_river_21, field="M2",
expression="!RASTERVALU!+0.4", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

        # Process: Calculate Field (3) (Calculate Field) (management)
        Extract_river_21_3_ =
arcpy.management.CalculateField(in_table=Extract_river_21_2_, field="M3",
expression="!RASTERVALU! + 0.6", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

        # Process: Calculate Field (4) (Calculate Field) (management)
        Extract_river_21_4_ =
arcpy.management.CalculateField(in_table=Extract_river_21_3_, field="M4",
expression="!RASTERVALU! + 0.8", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

        # Process: Calculate Field (5) (Calculate Field) (management)
        Extract_river_21_5_ =
arcpy.management.CalculateField(in_table=Extract_river_21_4_, field="M5",
expression="!RASTERVALU!+ 1", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

        # Process: Calculate Field (6) (Calculate Field) (management)
        Extract_river_21_6_ =
arcpy.management.CalculateField(in_table=Extract_river_21_5_, field="M6",
expression="!RASTERVALU! + 1.2", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]
```

```
        # Process: Calculate Field (7) (Calculate Field) (management)
        Extract_river_21_7_ = 
arcpy.management.CalculateField(in_table=Extract_river_21_6_, field="M7", 
expression="!RASTERVALU! + 1.4", expression_type="PYTHON3", 
code_block="", field_type="TEXT")[0]

        # Process: Calculate Field (8) (Calculate Field) (management)
        Extract_river_21_8_ = 
arcpy.management.CalculateField(in_table=Extract_river_21_7_, field="M8", 
expression="!RASTERVALU! + 1.6", expression_type="PYTHON3", 
code_block="", field_type="TEXT")[0]

        # Process: Calculate Field (9) (Calculate Field) (management)
        Extract_river_21_9_ = 
arcpy.management.CalculateField(in_table=Extract_river_21_8_, field="M9", 
expression="!RASTERVALU! + 1.8", expression_type="PYTHON3", 
code_block="", field_type="TEXT")[0]

        # Process: Calculate Field (10) (Calculate Field) (management)
        Extract_river_21_10_ = 
arcpy.management.CalculateField(in_table=Extract_river_21_9_, 
field="M10", expression="!RASTERVALU! + 2", expression_type="PYTHON3", 
code_block="", field_type="TEXT")[0]

        # Process: IDW (IDW) (3d)
        Idw_Extract_M1 = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third 
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M1"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif, 
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope 
Third Approach\Bank Slope Third Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third 
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_, 
z_field="M1", out_raster=Idw_Extract_M1, cell_size="1", power=5, 
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M1 = arcpy.Raster(Idw_Extract_M1)

        # Process: Raster Calculator (Raster Calculator) (ia)
        Idw_M1_boundary_tif = "E:\\GIS Automation\\Bank 
Slope_3\\Site_2\\IDW_M\\Idw_M1_boundary.tif"
        Raster_Calculator = Idw_M1_boundary_tif
        Idw_M1_boundary_tif =  SetNull(Idw_Extract_M1 
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M1)
        Idw_M1_boundary_tif.save(Raster_Calculator)


        # Process: Raster Domain (Raster Domain) (3d)
        Idw_M1_domain_shp = "E:\\GIS Automation\\Bank 
Slope_3\\Site_2\\Domain\\Idw_M1_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M1_boundary_tif, 
out_feature_class=Idw_M1_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (2) (IDW) (3d)
        Idw_Extract_M2 = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third 
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M2"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif, 
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope 
Third Approach\Bank Slope Third Approach.gdb", 
```

```
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_, z_field="M2", out_raster=Idw_Extract_M2, cell_size="1", power=5, search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M2 = arcpy.Raster(Idw_Extract_M2)

        # Process: Raster Calculator (2) (Raster Calculator) (ia)
        Idw_M2_boundary_tif = "E:\\GIS Automation\\Bank Slope_3\\Site_2\\IDW_M\\Idw_M2_boundary.tif"
        Raster_Calculator_2_ = Idw_M2_boundary_tif
        Idw_M2_boundary_tif =  SetNull(Idw_Extract_M2 <Gaulfoss_FullTerrain_tif ,Idw_Extract_M2)
        Idw_M2_boundary_tif.save(Raster_Calculator_2_)


        # Process: Raster Domain (2) (Raster Domain) (3d)
        Idw_M2_domain_shp = "E:\\GIS Automation\\Bank Slope_3\\Site_2\\Domain\\Idw_M2_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M2_boundary_tif, out_feature_class=Idw_M2_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (3) (IDW) (3d)
        Idw_Extract_M3 = "C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M3"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif, scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third Approach\Bank Slope Third Approach.gdb", workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_, z_field="M3", out_raster=Idw_Extract_M3, cell_size="1", power=5, search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M3 = arcpy.Raster(Idw_Extract_M3)

        # Process: Raster Calculator (3) (Raster Calculator) (ia)
        Idw_M3_boundary_tif = "E:\\GIS Automation\\Bank Slope_3\\Site_2\\IDW_M\\Idw_M3_boundary.tif"
        Raster_Calculator_3_ = Idw_M3_boundary_tif
        Idw_M3_boundary_tif =  SetNull(Idw_Extract_M3 <Gaulfoss_FullTerrain_tif ,Idw_Extract_M3)
        Idw_M3_boundary_tif.save(Raster_Calculator_3_)


        # Process: Raster Domain (3) (Raster Domain) (3d)
        Idw_M3_domain_shp = "E:\\GIS Automation\\Bank Slope_3\\Site_2\\Domain\\Idw_M3_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M3_boundary_tif, out_feature_class=Idw_M3_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (4) (IDW) (3d)
        Idw_Extract_M4 = "C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M4"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif, scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third Approach\Bank Slope Third Approach.gdb", workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
```

```
z_field="M4", out_raster=Idw_Extract_M4, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M4 = arcpy.Raster(Idw_Extract_M4)

        # Process: Raster Calculator (4) (Raster Calculator) (ia)
        Idw_M4_boundary_tif = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\IDW_M\\Idw_M4_boundary.tif"
        Raster_Calculator_4_ = Idw_M4_boundary_tif
        Idw_M4_boundary_tif =  SetNull(Idw_Extract_M4
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M4)
        Idw_M4_boundary_tif.save(Raster_Calculator_4_)


        # Process: Raster Domain (4) (Raster Domain) (3d)
        Idw_M4_domain_shp = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\Domain\\Idw_M4_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M4_boundary_tif,
out_feature_class=Idw_M4_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (5) (IDW) (3d)
        Idw_Extract_M5 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M5"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
z_field="M5", out_raster=Idw_Extract_M5, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M5 = arcpy.Raster(Idw_Extract_M5)

        # Process: Raster Calculator (5) (Raster Calculator) (ia)
        Idw_M5_boundary_tif = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\IDW_M\\Idw_M5_boundary.tif"
        Raster_Calculator_5_ = Idw_M5_boundary_tif
        Idw_M5_boundary_tif =  SetNull(Idw_Extract_M5
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M5)
        Idw_M5_boundary_tif.save(Raster_Calculator_5_)


        # Process: Raster Domain (5) (Raster Domain) (3d)
        Idw_M5_domain_shp = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\Domain\\Idw_M5_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M5_boundary_tif,
out_feature_class=Idw_M5_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (6) (IDW) (3d)
        Idw_Extract_M6 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M6"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
z_field="M6", out_raster=Idw_Extract_M6, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M6 = arcpy.Raster(Idw_Extract_M6)
```

```
        # Process: Raster Calculator (6) (Raster Calculator) (ia)
        Idw_M6_boundary_tif = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\IDW_M\\Idw_M6_boundary.tif"
        Raster_Calculator_6_ = Idw_M6_boundary_tif
        Idw_M6_boundary_tif =  SetNull(Idw_Extract_M6
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M6)
        Idw_M6_boundary_tif.save(Raster_Calculator_6_)


        # Process: Raster Domain (6) (Raster Domain) (3d)
        Idw_M6_domain_shp = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\Domain\\Idw_M6_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M6_boundary_tif,
out_feature_class=Idw_M6_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (7) (IDW) (3d)
        Idw_Extract_M7 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M7"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
z_field="M7", out_raster=Idw_Extract_M7, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M7 = arcpy.Raster(Idw_Extract_M7)

        # Process: Raster Calculator (7) (Raster Calculator) (ia)
        Idw_M7_boundary_tif = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\IDW_M\\Idw_M7_boundary.tif"
        Raster_Calculator_7_ = Idw_M7_boundary_tif
        Idw_M7_boundary_tif =  SetNull(Idw_Extract_M7
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M7)
        Idw_M7_boundary_tif.save(Raster_Calculator_7_)


        # Process: Raster Domain (7) (Raster Domain) (3d)
        Idw_M7_domain_shp = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\Domain\\Idw_M7_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M7_boundary_tif,
out_feature_class=Idw_M7_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (8) (IDW) (3d)
        Idw_Extract_M8 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M8"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
z_field="M8", out_raster=Idw_Extract_M8, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M8 = arcpy.Raster(Idw_Extract_M8)

        # Process: Raster Calculator (8) (Raster Calculator) (ia)
        Idw_M8_boundary_tif = "E:\\GIS Automation\\Bank
```

```
Slope_3\\Site_2\\IDW_M\\Idw_M8_boundary.tif"
        Raster_Calculator_8_ = Idw_M8_boundary_tif
        Idw_M8_boundary_tif =  SetNull(Idw_Extract_M8
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M8)
        Idw_M8_boundary_tif.save(Raster_Calculator_8_)


        # Process: Raster Domain (8) (Raster Domain) (3d)
        Idw_M8_domain_shp = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\Domain\\Idw_M8_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M8_boundary_tif,
out_feature_class=Idw_M8_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (9) (IDW) (3d)
        Idw_Extract_M9 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M9"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
z_field="M9", out_raster=Idw_Extract_M9, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M9 = arcpy.Raster(Idw_Extract_M9)

        # Process: Raster Calculator (9) (Raster Calculator) (ia)
        Idw_M9_boundary_tif = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\IDW_M\\Idw_M9_boundary.tif"
        Raster_Calculator_9_ = Idw_M9_boundary_tif
        Idw_M9_boundary_tif =  SetNull(Idw_Extract_M9
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M9)
        Idw_M9_boundary_tif.save(Raster_Calculator_9_)


        # Process: Raster Domain (9) (Raster Domain) (3d)
        Idw_M9_domain_shp = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\Domain\\Idw_M9_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M9_boundary_tif,
out_feature_class=Idw_M9_domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (10) (IDW) (3d)
        Idw_Extract_M10 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Idw_Extract_M10"
        with arcpy.EnvManager(extent=Gaulfoss_FullTerrain_tif,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Extract_river_21_10_,
z_field="M10", out_raster=Idw_Extract_M10, cell_size="1", power=5,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_Extract_M10 = arcpy.Raster(Idw_Extract_M10)

        # Process: Raster Calculator (10) (Raster Calculator) (ia)
        Idw_M10_boundary_tif = "E:\\GIS Automation\\Bank
Slope_3\\Site_2\\IDW_M\\Idw_M10_boundary.tif"
        Raster_Calculator_10_ = Idw_M10_boundary_tif
        Idw_M10_boundary_tif =  SetNull(Idw_Extract_M10
```

```
<Gaulfoss_FullTerrain_tif ,Idw_Extract_M10)
        Idw_M10_boundary_tif.save(Raster_Calculator_10_)


        # Process: Raster Domain (10) (Raster Domain) (3d)
        Idw_M10_domain_shp = "E:\\GIS Automation\\Bank 
Slope_3\\Site_2\\Domain\\Idw_M10_domain.shp"
        arcpy.ddd.RasterDomain(in_raster=Idw_M10_boundary_tif, 
out_feature_class=Idw_M10_domain_shp, out_geometry_type="POLYGON")

if __name__ == '__main__':
    Model()
```

b) For saving the elevation of layers in tables

```
# -*- coding: utf-8 -*-
"""
Generated by ArcGIS ModelBuilder on : 2021-03-08 15:47:08
"""
import arcpy
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia import *
#from arcpy.ia i

#def Model():  # Model

# To allow overwriting outputs change overwriteOutput option to True.
arcpy.env.overwriteOutput = True

# Check out any necessary licenses.
arcpy.CheckOutExtension("spatial")
arcpy.CheckOutExtension("3D")
arcpy.CheckOutExtension("ImageAnalyst")
i = 7
arcpy.ImportToolbox(r"c:\program 
files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management 
Tools.tbx")
# Model Environment settings
with 
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third 
Approach\Bank Slope Third Approach.gdb"):
    river_3_shp = "E:\\GIS Automation\\Bank Slope_2\\Site_" + str(i) + 
r"\river_" + str(i) + r".shp"
    Gaulfoss_FullTerrain_tif = arcpy.Raster(r"E:\LiDAR River 
Volume\Tokke\Final Results\Merged Terrain\Tokke_FullTerrain.tif")
    Bank_Slope_Third_Approach_gdb = r"E:\GIS Automation\Bank 
Slope_3\Site_" + str(i) + r"\Output"

    # Process: Generate Points Along Lines (Generate Points Along Lines) 
(management)
```

```
    river_1_GeneratePointsAlongLines1_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\river_3_GeneratePointsAlongLines2.shp"
    with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):

arcpy.management.GeneratePointsAlongLines(Input_Features=river_3_shp,
Output_Feature_Class=river_1_GeneratePointsAlongLines1_shp,
Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
Include_End_Points="")

    # Process: Extract Values to Points (Extract Values to Points) (sa)
    Output_point_features =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Third
Approach\\Bank Slope Third Approach.gdb\\Extract_river_31"
    with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):

arcpy.sa.ExtractValuesToPoints(in_point_features=river_1_GeneratePointsAl
ongLines1_shp, in_raster=Gaulfoss_FullTerrain_tif,
out_point_features=Output_point_features, interpolate_values="NONE",
add_attributes="VALUE_ONLY")

    # Process: Add Field (Add Field) (management)
    Extract_river_11 =
arcpy.management.AddField(in_table=Output_point_features,
field_name="M1", field_type="DOUBLE", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

    # Process: Add Field (2) (Add Field) (management)
    Extract_river_11_2_ =
arcpy.management.AddField(in_table=Extract_river_11, field_name="M2",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (3) (Add Field) (management)
    Extract_river_11_3_ =
arcpy.management.AddField(in_table=Extract_river_11_2_, field_name="M3",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (4) (Add Field) (management)
    Extract_river_11_4_ =
arcpy.management.AddField(in_table=Extract_river_11_3_, field_name="M4",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (5) (Add Field) (management)
    Extract_river_11_5_ =
arcpy.management.AddField(in_table=Extract_river_11_4_, field_name="M5",
```

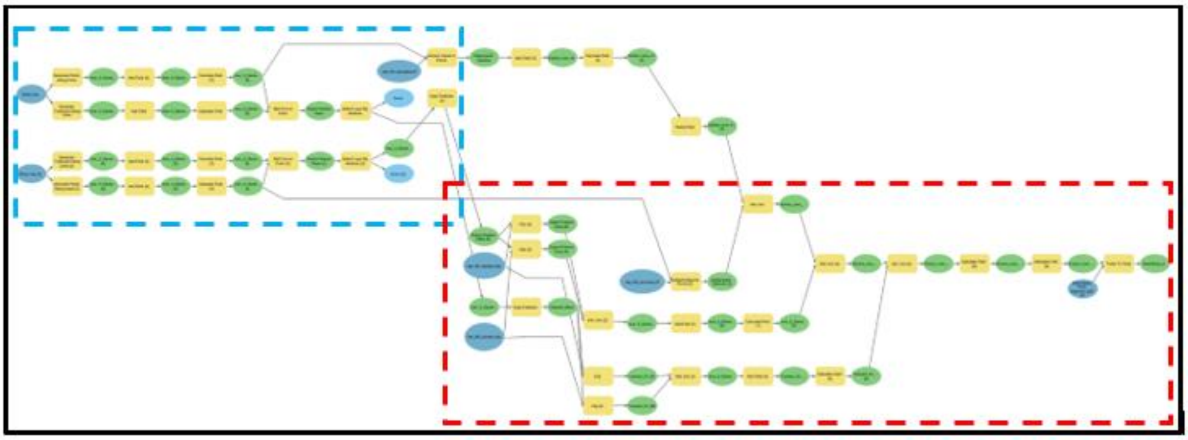
```
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (6) (Add Field) (management)
    Extract_river_11_6_ =
arcpy.management.AddField(in_table=Extract_river_11_5_, field_name="M6",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (7) (Add Field) (management)
    Extract_river_11_7_ =
arcpy.management.AddField(in_table=Extract_river_11_6_, field_name="M7",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (8) (Add Field) (management)
    Extract_river_11_8_ =
arcpy.management.AddField(in_table=Extract_river_11_7_, field_name="M8",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (9) (Add Field) (management)
    Extract_river_11_9_ =
arcpy.management.AddField(in_table=Extract_river_11_8_, field_name="M9",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Add Field (10) (Add Field) (management)
    Extract_river_11_10_ =
arcpy.management.AddField(in_table=Extract_river_11_9_, field_name="M10",
field_type="DOUBLE", field_precision=None, field_scale=None,
field_length=None, field_alias="", field_is_nullable="NULLABLE",
field_is_required="NON_REQUIRED", field_domain="")[0]

    # Process: Calculate Field (Calculate Field) (management)
    Extract_river_21 =
arcpy.management.CalculateField(in_table=Extract_river_11_10_,
field="M1", expression="!RASTERVALU!+1", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

    # Process: Calculate Field (2) (Calculate Field) (management)
    Extract_river_21_2_ =
arcpy.management.CalculateField(in_table=Extract_river_21, field="M2",
expression="!RASTERVALU!+2", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (3) (Calculate Field) (management)
    Extract_river_21_3_ =
arcpy.management.CalculateField(in_table=Extract_river_21_2_, field="M3",
expression="!RASTERVALU! + 3", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (4) (Calculate Field) (management)
    Extract_river_21_4_ =
arcpy.management.CalculateField(in_table=Extract_river_21_3_, field="M4",
expression="!RASTERVALU! + 4", expression_type="PYTHON3", code_block="",
```

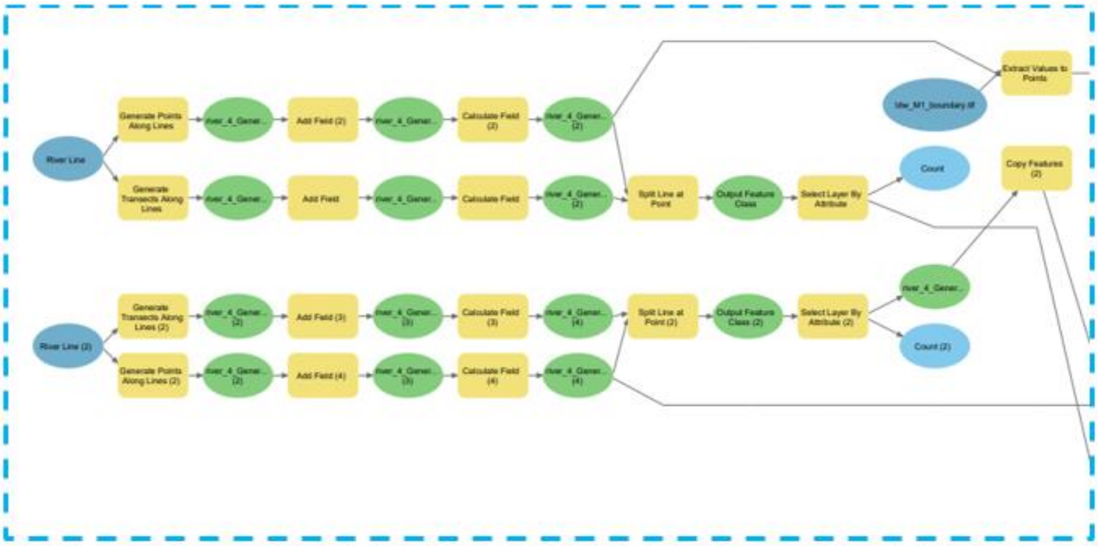
```
field_type="TEXT")[0]

    # Process: Calculate Field (5) (Calculate Field) (management)
    Extract_river_21_5_ =
arcpy.management.CalculateField(in_table=Extract_river_21_4_, field="M5",
expression="!RASTERVALU!+ 5", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (6) (Calculate Field) (management)
    Extract_river_21_6_ =
arcpy.management.CalculateField(in_table=Extract_river_21_5_, field="M6",
expression="!RASTERVALU! + 6", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (7) (Calculate Field) (management)
    Extract_river_21_7_ =
arcpy.management.CalculateField(in_table=Extract_river_21_6_, field="M7",
expression="!RASTERVALU! + 7", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (8) (Calculate Field) (management)
    Extract_river_21_8_ =
arcpy.management.CalculateField(in_table=Extract_river_21_7_, field="M8",
expression="!RASTERVALU! + 8", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (9) (Calculate Field) (management)
    Extract_river_21_9_ =
arcpy.management.CalculateField(in_table=Extract_river_21_8_, field="M9",
expression="!RASTERVALU! + 9", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]

    # Process: Calculate Field (10) (Calculate Field) (management)
    Extract_river_21_10_ =
arcpy.management.CalculateField(in_table=Extract_river_21_9_,
field="M10", expression="!RASTERVALU! + 10", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]


    # Process: Table To Table (Table To Table) (conversion)
    with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Third Approach\Bank Slope Third Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Third
Approach\Bank Slope Third Approach.gdb"):
        table_naming = r"Interval Levels_" + str(i) + r".csv"
        table_10points =
arcpy.conversion.TableToTable(in_rows=Extract_river_21_10_,
out_path=Bank_Slope_Third_Approach_gdb, out_name= table_naming,
where_clause="", field_mapping="M1 \"M1\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M1,-1,-
1;M2 \"M2\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M2,-1,-
1;M3 \"M3\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M3,-1,-
1;M4 \"M4\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M4,-1,-
```

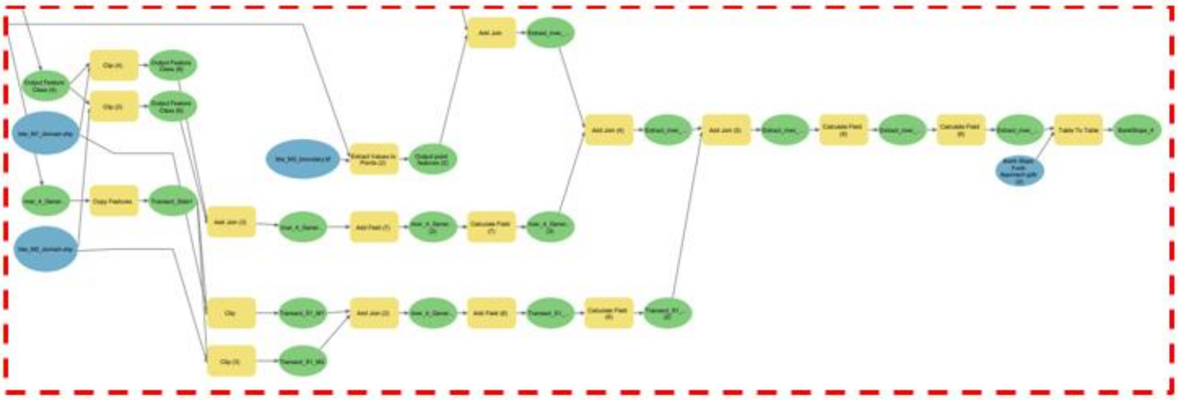
```
1;M5 \"M5\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M5,-1,-
1;M6 \"M6\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M6,-1,-
1;M7 \"M7\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M7,-1,-
1;M8 \"M8\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M8,-1,-
1;M9 \"M9\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M9,-1,-
1;M10 \"M10\" true true false 0 Double 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Third Approach\\Bank Slope Third Approach.gdb\\Extract_river_31,M10,-1,-
1", config_keyword="")[0]

#if __name__ == '__main__':
    #Model()
```

Approach 3

1. ModelBuilder chart





2. Python code for the automation

```
# -*- coding: utf-8 -*-
"""
Generated by ArcGIS ModelBuilder on : 2021-03-13 18:12:23
"""
import arcpy

#def Model():  # Model

# To allow overwriting outputs change overwriteOutput option to True.
arcpy.env.overwriteOutput = True

# Check out any necessary licenses.
arcpy.CheckOutExtension("spatial")
var1 = [3]  # Site Number
var2 = [2, 3, 4, 5, 6, 7, 8, 9, 10]  # Number of the Zone of the Ten
Zones (M)
var4 = [1, 2, 3, 4, 5, 6, 7]
var3 = [850, 550, 2000, 1500, 1500, 800, 500, 500, 750]  # The Suitable
length of Transect in meters
arcpy.ImportToolbox(r"c:\program
files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management
Tools.tbx")
for i in var1:
    for e in var4:
        for j in var2:
            # Model Environment settings
            m = j - 1
            n = j
            with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                River_Line = r"E:\GIS Automation\River Lines Backup" +
r"\river_" + str(i) + r".shp"
                River_Line_2_ = r"E:\GIS Automation\River Lines Backup" +
r"\river_" + str(i) + r".shp"
                WSE_L_Path = r"E:\GIS Automation\Bank Slope_5\Site_" +
str(i) + r"\IDW\Idw_M" + str(m) + r"_boundary.tif"
                WSE_U_Path = r"E:\GIS Automation\Bank Slope_5\Site_" +
str(i) + r"\IDW\Idw_M" + str(n) + r"_boundary.tif"
                Inund_L_Path = r"E:\GIS Automation\Bank Slope_5\Site_" +
str(i) + r"\Domain_Clip\IDW_boundary_Domain_" + str(e) + str(m) + r".shp"
                Inund_U_Path = r"E:\GIS Automation\Bank Slope_5\Site_" +
str(i) + r"\Domain_Clip\IDW_boundary_Domain_" + str(e) + str(n) + r".shp"
                Output = r"E:\GIS Automation\Bank Slope_5\Site_" + str(i)
+ r"\Output"
                Idw_M1_boundary_tif = WSE_L_Path
                Idw_M2_boundary_tif = WSE_U_Path
                Idw_M2_domain_shp = Inund_U_Path
                Idw_M1_domain_shp = Inund_L_Path
                Bank_Slope_Forth_Approach_gdb_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb"

                # Process: Generate Transects Along Lines (Generate
Transects Along Lines) (management)
                river_4_GenerateTransectsAlongLines1_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
```

```
Approach\\river_4_GenerateTransectsAlongLines6.shp"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.GenerateTransectsAlongLines(in_features=River_Line,
out_feature_class=river_4_GenerateTransectsAlongLines1_shp, interval="50
Meters", transect_length="{} Meters".format(var3[i-1]),
include_ends="NO_END_POINTS")

                # Process: Add Field (Add Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlongLines2_shp =
arcpy.management.AddField(in_table=river_4_GenerateTransectsAlongLines1_s
hp, field_name="Cop_FID1", field_type="LONG", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

                # Process: Calculate Field (Calculate Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlongLines2_shp_2_ =
arcpy.management.CalculateField(in_table=river_4_GenerateTransectsAlongLi
nes2_shp, field="Cop_FID1", expression="!FID!+1",
expression_type="PYTHON3", code_block="", field_type="TEXT")[0]

                # Process: Generate Points Along Lines (Generate Points
Along Lines) (management)
                river_4_GeneratePointsAlongLines1_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\river_4_GeneratePointsAlongLines6.shp"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.GeneratePointsAlongLines(Input_Features=River_Line,
Output_Feature_Class=river_4_GeneratePointsAlongLines1_shp,
Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
Include_End_Points="")

                # Process: Add Field (2) (Add Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GeneratePointsAlongLines2_shp =
arcpy.management.AddField(in_table=river_4_GeneratePointsAlongLines1_shp,
field_name="Cop_FID2", field_type="LONG", field_precision=None,
```

```
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

                # Process: Calculate Field (2) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GeneratePointsAlongLines2_shp_2_ =
arcpy.management.CalculateField(in_table=river_4_GeneratePointsAlongLines
2_shp, field="Cop_FID2", expression="!FID!+1", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

                # Process: Split Line at Point (Split Line at Point)
(management)
                Output_Feature_Class =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\river_4_GenerateTransectsAlo"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.SplitLineAtPoint(in_features=river_4_GenerateTransectsAl
ongLines2_shp_2_,
point_features=river_4_GeneratePointsAlongLines2_shp_2_,
out_feature_class=Output_Feature_Class, search_radius="")

                # Process: Select Layer By Attribute (Select Layer By
Attribute) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlo, Count =
arcpy.management.SelectLayerByAttribute(in_layer_or_view=Output_Feature_C
lass, selection_type="NEW_SELECTION", where_clause="MOD(OBJECTID,2)=0",
invert_where_clause="")

                # Process: Generate Transects Along Lines (2) (Generate
Transects Along Lines) (management)
                river_4_GenerateTransectsAlongLines1_shp_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\river_4_GenerateTransectsAlongLines1.shp"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.GenerateTransectsAlongLines(in_features=River_Line_2_,
out_feature_class=river_4_GenerateTransectsAlongLines1_shp_2_,
interval="50 Meters", transect_length="{} Meters".format(var3[i-1]),
include_ends="NO_END_POINTS")

                # Process: Add Field (3) (Add Field) (management)
```

```
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlongLines1_shp_3_ =
arcpy.management.AddField(in_table=river_4_GenerateTransectsAlongLines1_shp_2_, field_name="Cop_FID3", field_type="LONG", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

                # Process: Calculate Field (3) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlongLines1_shp_4_ =
arcpy.management.CalculateField(in_table=river_4_GenerateTransectsAlongLines1_shp_3_, field="Cop_FID3", expression="!FID!+1",
expression_type="PYTHON3", code_block="", field_type="TEXT")[0]

                # Process: Generate Points Along Lines (2) (Generate
Points Along Lines) (management)
                river_4_GeneratePointsAlongLines1_shp_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\river_4_GeneratePointsAlongLines1.shp"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.GeneratePointsAlongLines(Input_Features=River_Line_2_,
Output_Feature_Class=river_4_GeneratePointsAlongLines1_shp_2_,
Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
Include_End_Points="")

                # Process: Add Field (4) (Add Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GeneratePointsAlongLines1_shp_3_ =
arcpy.management.AddField(in_table=river_4_GeneratePointsAlongLines1_shp_2_, field_name="Cop_FID4", field_type="LONG", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

                # Process: Calculate Field (4) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GeneratePointsAlongLines1_shp_4_ =
```

```
arcpy.management.CalculateField(in_table=river_4_GeneratePointsAlongLines
1_shp_3_, field="Cop_FID4", expression="!FID!+1",
expression_type="PYTHON3", code_block="", field_type="TEXT")[0]

                # Process: Split Line at Point (2) (Split Line at Point)
(management)
                Output_Feature_Class_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\river_4_GenerateTransectsAlo1"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.SplitLineAtPoint(in_features=river_4_GenerateTransectsAl
ongLines1_shp_4_,
point_features=river_4_GeneratePointsAlongLines1_shp_4_,
out_feature_class=Output_Feature_Class_2_, search_radius="")

                # Process: Select Layer By Attribute (2) (Select Layer By
Attribute) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlo1, Count_2_ =
arcpy.management.SelectLayerByAttribute(in_layer_or_view=Output_Feature_C
lass_2_, selection_type="NEW_SELECTION", where_clause="MOD(OBJECTID,2) <>
0 ", invert_where_clause="")

                # Process: Extract Values to Points (Extract Values to
Points) (sa)
                Output_point_features =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\Extract_river_41"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.sa.ExtractValuesToPoints(in_point_features=river_4_GeneratePointsAl
ongLines2_shp_2_, in_raster=Idw_M1_boundary_tif,
out_point_features=Output_point_features, interpolate_values="NONE",
add_attributes="VALUE_ONLY")

                # Process: Add Field (5) (Add Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41 =
arcpy.management.AddField(in_table=Output_point_features,
field_name="Low_Elev", field_type="DOUBLE", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]
```

```
                # Process: Calculate Field (5) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_2_ =
arcpy.management.CalculateField(in_table=Extract_river_41,
field="Low_Elev", expression="!RASTERVALU!", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

                # Process: Delete Field (Delete Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_3_ =
arcpy.management.DeleteField(in_table=Extract_river_41_2_,
drop_field=["RASTERVALU"])[0]

                # Process: Extract Values to Points (2) (Extract Values
to Points) (sa)
                Output_point_features_2_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\Extract_river_42"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.sa.ExtractValuesToPoints(in_point_features=river_4_GeneratePointsAlongLines1_shp_4_, in_raster=Idw_M2_boundary_tif,
out_point_features=Output_point_features_2_, interpolate_values="NONE",
add_attributes="VALUE_ONLY")

                # Process: Add Join (Add Join) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_Layer =
arcpy.management.AddJoin(in_layer_or_view=Extract_river_41_3_,
in_field="Cop_FID2", join_table=Output_point_features_2_,
join_field="Cop_FID4", join_type="KEEP_ALL")[0]

                # Process: Copy Features (2) (Copy Features) (management)
                Output_Feature_Class_4_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth
Approach.gdb\\river_4_GenerateTransectsAlo1_CopyFeatures"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.CopyFeatures(in_features=river_4_GenerateTransectsAlo1,
```

```
out_feature_class=Output_Feature_Class_4_, config_keyword="",
spatial_grid_1=None, spatial_grid_2=None, spatial_grid_3=None)

                # Process: Clip (2) (Clip) (analysis)
                Output_Feature_Class_6_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\river_4_GenerateTransectsAlo2"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.analysis.Clip(in_features=Output_Feature_Class_4_,
clip_features=Idw_M2_domain_shp,
out_feature_class=Output_Feature_Class_6_, cluster_tolerance="")

                # Process: Clip (4) (Clip) (analysis)
                Output_Feature_Class_8_ =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\river_4_GenerateTransectsAlo4"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.analysis.Clip(in_features=Output_Feature_Class_4_,
clip_features=Idw_M1_domain_shp,
out_feature_class=Output_Feature_Class_8_, cluster_tolerance="")

                # Process: Add Join (3) (Add Join) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlo5 =
arcpy.management.AddJoin(in_layer_or_view=Output_Feature_Class_6_,
in_field="Cop_FID3", join_table=Output_Feature_Class_8_,
join_field="Cop_FID3", join_type="KEEP_ALL")[0]

                # Process: Add Field (7) (Add Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlo5_2_ =
arcpy.management.AddField(in_table=river_4_GenerateTransectsAlo5,
field_name="L2_Diff", field_type="DOUBLE", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

                # Process: Calculate Field (7) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
```

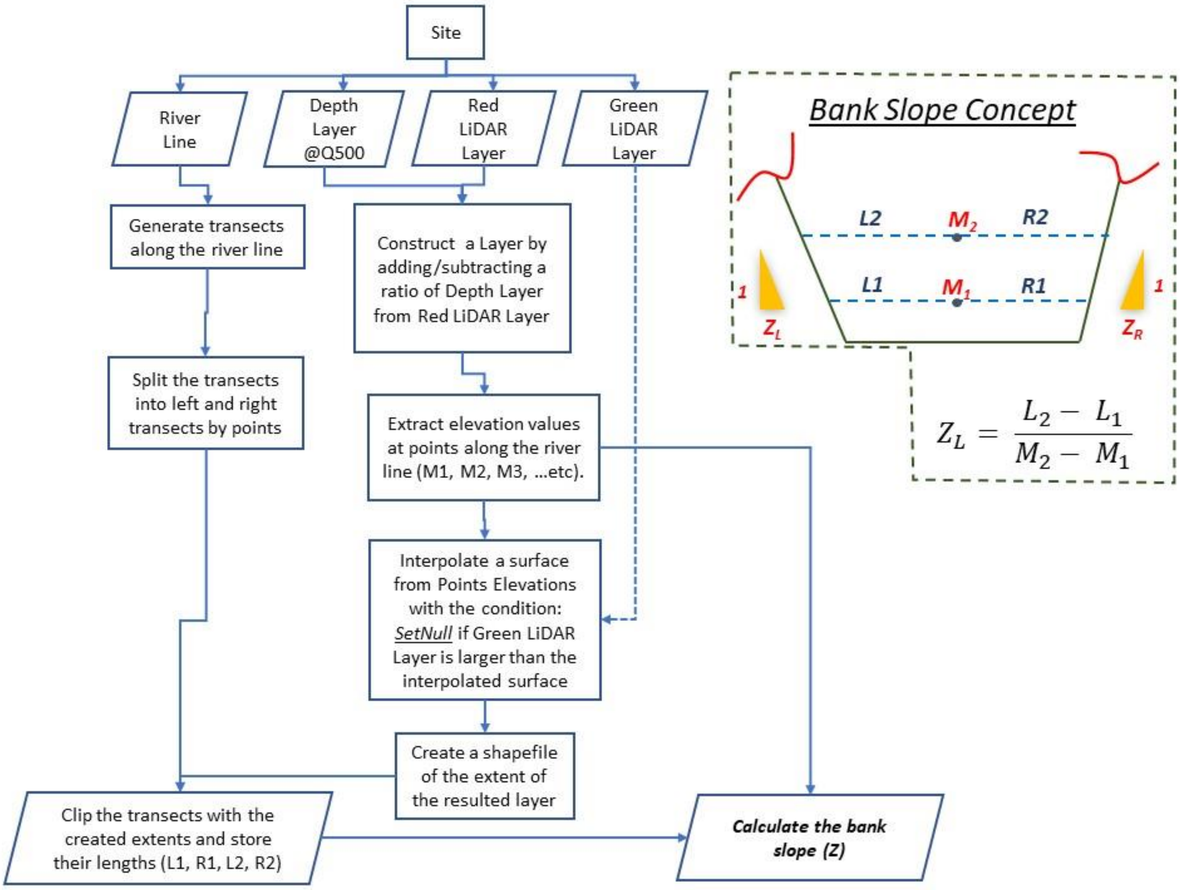
```
Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlo5_3_ =
arcpy.management.CalculateField(in_table=river_4_GenerateTransectsAlo5_2_
, field="L2_Diff",
expression="!river_4_GenerateTransectsAlo2.Shape_Length! -
!river_4_GenerateTransectsAlo4.Shape_Length! ",
expression_type="PYTHON3", code_block="", field_type="TEXT")[0]

                # Process: Add Join (4) (Add Join) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_Layer1 =
arcpy.management.AddJoin(in_layer_or_view=Extract_river_41_Layer,
in_field="Extract_river_41.Cop_FID2",
join_table=river_4_GenerateTransectsAlo5_3_,
join_field="river_4_GenerateTransectsAlo2.Cop_FID3",
join_type="KEEP_ALL")[0]

                # Process: Copy Features (Copy Features) (management)
                Transect_Side1 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\Transect_Side1"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):

arcpy.management.CopyFeatures(in_features=river_4_GenerateTransectsAlo,
out_feature_class=Transect_Side1, config_keyword="", spatial_grid_1=None,
spatial_grid_2=None, spatial_grid_3=None)

                # Process: Clip (Clip) (analysis)
                Transect_S1_M1 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\Transect_S1_M1"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    arcpy.analysis.Clip(in_features=Transect_Side1,
clip_features=Idw_M1_domain_shp, out_feature_class=Transect_S1_M1,
cluster_tolerance="")

                # Process: Clip (3) (Clip) (analysis)
                Transect_S1_M2 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Forth
Approach\\Bank Slope Forth Approach.gdb\\Transect_S1_M2"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth
Approach\Bank Slope Forth Approach.gdb"):
                    arcpy.analysis.Clip(in_features=Transect_Side1,
clip_features=Idw_M2_domain_shp, out_feature_class=Transect_S1_M2,
cluster_tolerance="")
```

```
                # Process: Add Join (2) (Add Join) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    river_4_GenerateTransectsAlo4 =
arcpy.management.AddJoin(in_layer_or_view=Transect_S1_M1,
in_field="Cop_FID1", join_table=Transect_S1_M2, join_field="Cop_FID1",
join_type="KEEP_ALL")[0]

                # Process: Add Field (6) (Add Field) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    Transect_S1_M1_Layer =
arcpy.management.AddField(in_table=river_4_GenerateTransectsAlo4,
field_name="L1_Diff", field_type="DOUBLE", field_precision=None,
field_scale=None, field_length=None, field_alias="",
field_is_nullable="NULLABLE", field_is_required="NON_REQUIRED",
field_domain="")[0]

                # Process: Calculate Field (6) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    Transect_S1_M1_Layer_2_ =
arcpy.management.CalculateField(in_table=Transect_S1_M1_Layer,
field="Transect_S1_M1.L1_Diff",
expression="!Transect_S1_M2.Shape_Length!-!Transect_S1_M1.Shape_Length!",
expression_type="PYTHON3", code_block="", field_type="TEXT")[0]

                # Process: Add Join (5) (Add Join) (management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_Layer2 =
arcpy.management.AddJoin(in_layer_or_view=Extract_river_41_Layer1,
in_field="Extract_river_41.Cop_FID2", join_table=Transect_S1_M1_Layer_2_,
join_field="Transect_S1_M1.Cop_FID1", join_type="KEEP_ALL")[0]

                # Process: Calculate Field (8) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_Layer3 =
arcpy.management.CalculateField(in_table=Extract_river_41_Layer2,
field="Z1",
expression="!Transect_S1_M1.L1_Diff!/(!Extract_river_42.RASTERVALU! -
!Extract_river_41.Low_Elev!)", expression_type="PYTHON3", code_block="",
field_type="TEXT")[0]
```

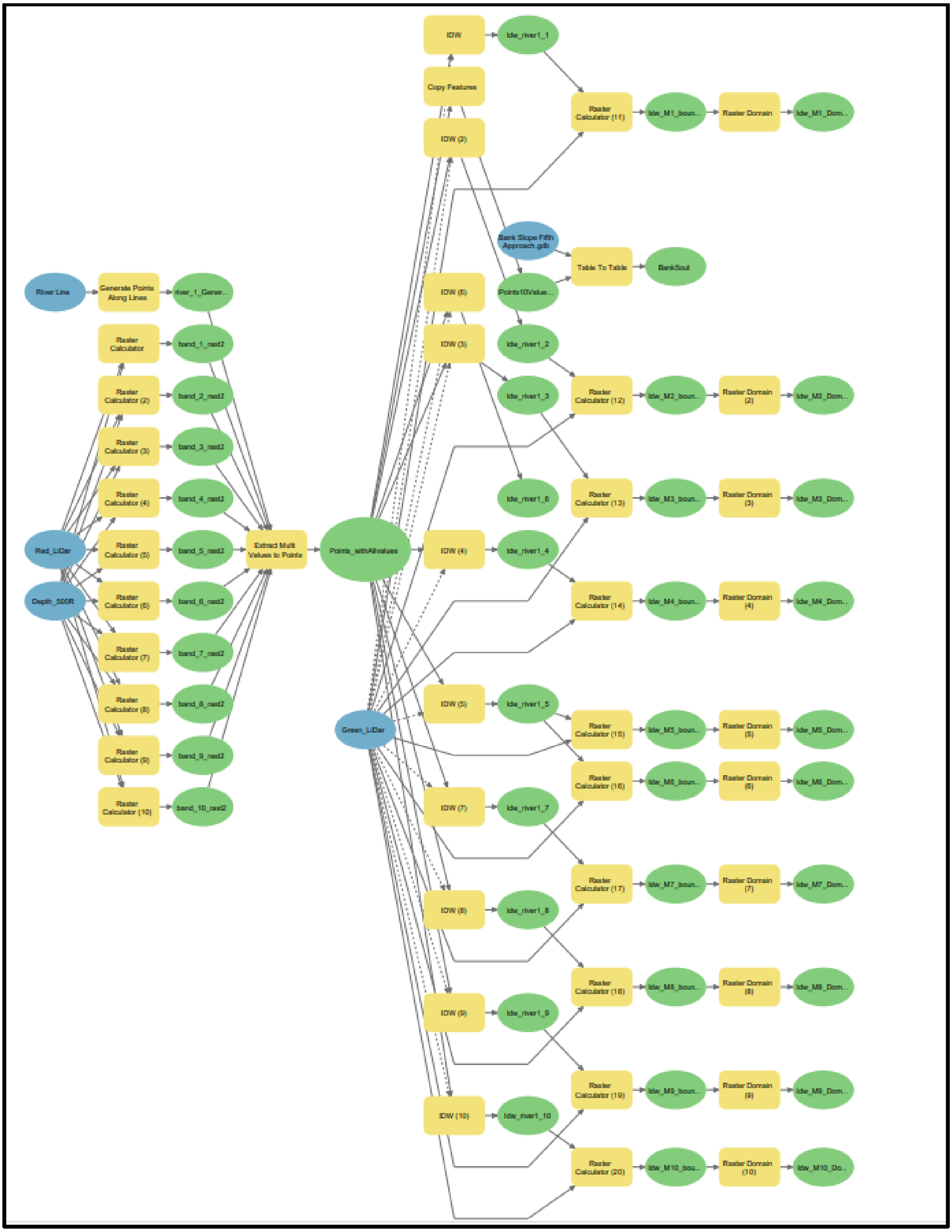
```
                # Process: Calculate Field (9) (Calculate Field)
(management)
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    Extract_river_41_Layer4 =
arcpy.management.CalculateField(in_table=Extract_river_41_Layer3,
field="Z2",
expression="!river_4_GenerateTransectsAlo2.L2_Diff!/(!Extract_river_42.RASTERVALU! - !Extract_river_41.Low_Elev!)", expression_type="PYTHON3",
code_block="", field_type="TEXT")[0]

                # Process: Table To Table (Table To Table) (conversion)
                FFF = "BankSideSlopes6_" + str(i) + "_" + str(e) + "_" +
str(m) + ".csv"
                with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Forth Approach\Bank Slope Forth Approach.gdb"):
                    BankSlope_4 =
arcpy.conversion.TableToTable(in_rows=Extract_river_41_Layer4, out_path=
Output, out_name= FFF, where_clause="", field_mapping="Cop_FID2
\"Cop_FID2\" true true false 4 Long 0
0,First,#,Extract_river_41_Layer4,Extract_river_41.Cop_FID2,-1,-1;Z1
\"Z1\" true true false 512 Text 0
0,First,#,Extract_river_41_Layer4,Extract_river_41.Z1,0,512;Z2 \"Z2\"
true true false 512 Text 0
0,First,#,Extract_river_41_Layer4,Extract_river_41.Z2,0,512",
config_keyword="")[0]
    print("You Finished " + str(i) + " Now ^^")
#if __name__ == '__main__':
    #Model()
```

Approach 4 (The final developed approach)

1. Schematic flow chart of the concept



2. ModelBuilder Chart



3. Python code for automation

```
# -*- coding: utf-8 -*-
"""
Generated by ArcGIS ModelBuilder on : 2021-03-29 19:43:09
"""
import arcpy
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *
from arcpy.ia import *

def Model1():  # Model 1

    # To allow overwriting outputs change overwriteOutput option to True.
    arcpy.env.overwriteOutput = False

    # Check out any necessary licenses.
    arcpy.CheckOutExtension("spatial")
    arcpy.CheckOutExtension("ImageAnalyst")
    arcpy.CheckOutExtension("3D")

    arcpy.ImportToolbox(r"c:\program
files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management
Tools.tbx")
    # Model Environment settings
    with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
        River_Line = "E:\\GIS Automation\\River Lines
Backup\\river_1.shp"
        Red_LiDar = arcpy.Raster("E:\\GIS Automation\\Bank
Slope_5\\Site_1\\RED LIDAR\\RED_LIDAR_1.tif")
        Depth_500R = arcpy.Raster("E:\\GIS Automation\\Bank
Slope_2\\Site_1\\Depth\\Depth_R500.tif")
        Bank_Slope_Fifth_Approach_gdb =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb"
        Green_LiDar = arcpy.Raster("E:\\GIS Automation\\Full Terrain
Models\\Terrain_1.tif")
```

```
        # Process: Generate Points Along Lines (Generate Points Along
Lines) (management)
        river_1_GeneratePointsAlongLines4_shp =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\river_1_GeneratePointsAlongLines7.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):

arcpy.management.GeneratePointsAlongLines(Input_Features=River_Line,
Output_Feature_Class=river_1_GeneratePointsAlongLines4_shp,
Point_Placement="DISTANCE", Distance="50 Meters", Percentage=None,
Include_End_Points="")

        # Process: Raster Calculator (Raster Calculator) (ia)
        band_1_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_1_rast2"
        Raster_Calculator = band_1_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_1_rast2 =  Red_LiDar - 0.1 * Depth_500R
            band_1_rast2.save(Raster_Calculator)


        # Process: Raster Calculator (2) (Raster Calculator) (ia)
        band_2_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_2_rast2"
        Raster_Calculator_2_ = band_2_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_2_rast2 =  Red_LiDar + 0.2 * Depth_500R
            band_2_rast2.save(Raster_Calculator_2_)


        # Process: Raster Calculator (3) (Raster Calculator) (ia)
        band_3_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_3_rast2"
        Raster_Calculator_3_ = band_3_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_3_rast2 =  Red_LiDar + 0.3 * Depth_500R
            band_3_rast2.save(Raster_Calculator_3_)


        # Process: Raster Calculator (4) (Raster Calculator) (ia)
        band_4_rast2 =
```

```
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_4_rast2"
        Raster_Calculator_4_ = band_4_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_4_rast2 =  Red_LiDar + 0.4 * Depth_500R
            band_4_rast2.save(Raster_Calculator_4_)


        # Process: Raster Calculator (5) (Raster Calculator) (ia)
        band_5_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_5_rast2"
        Raster_Calculator_5_ = band_5_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_5_rast2 =  Red_LiDar + 0.5 * Depth_500R
            band_5_rast2.save(Raster_Calculator_5_)


        # Process: Raster Calculator (6) (Raster Calculator) (ia)
        band_6_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_6_rast2"
        Raster_Calculator_6_ = band_6_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_6_rast2 =  Red_LiDar + 0.6 * Depth_500R
            band_6_rast2.save(Raster_Calculator_6_)


        # Process: Raster Calculator (7) (Raster Calculator) (ia)
        band_7_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_7_rast2"
        Raster_Calculator_7_ = band_7_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_7_rast2 =  Red_LiDar  + 0.7 * Depth_500R
            band_7_rast2.save(Raster_Calculator_7_)


        # Process: Raster Calculator (8) (Raster Calculator) (ia)
        band_8_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_8_rast2"
        Raster_Calculator_8_ = band_8_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
```

```
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_8_rast2 =  Red_LiDar + 0.8 * Depth_500R
            band_8_rast2.save(Raster_Calculator_8_)


        # Process: Raster Calculator (9) (Raster Calculator) (ia)
        band_9_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_9_rast2"
        Raster_Calculator_9_ = band_9_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_9_rast2 =  Red_LiDar  + 0.9 * Depth_500R
            band_9_rast2.save(Raster_Calculator_9_)


        # Process: Raster Calculator (10) (Raster Calculator) (ia)
        band_10_rast2 =
"c:\\Users\\moawadal\\documents\\ArcGIS\\Projects\\bank slope fifth
approach\\bank slope fifth approach.gdb\\band_10_rast2"
        Raster_Calculator_10_ = band_10_rast2
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            band_10_rast2 =  Red_LiDar + 1.2 * Depth_500R
            band_10_rast2.save(Raster_Calculator_10_)


        # Process: Extract Multi Values to Points (Extract Multi Values
to Points) (sa)
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Points_withAllvalues =
arcpy.sa.ExtractMultiValuesToPoints(in_point_features=river_1_GeneratePoi
ntsAlongLines4_shp, in_rasters=[[band_1_rast2, "band_1_r_2"],
[band_2_rast2, "band_2_r_2"], [band_3_rast2, "band_3_r_2"],
[band_4_rast2, "band_4_r_2"], [band_5_rast2, "band_5_r_2"],
[band_6_rast2, "band_6_r_2"], [band_7_rast2, "band_7_r_2"],
[band_8_rast2, "band_8_r_2"], [band_9_rast2, "band_9_r_2"],
[band_10_rast2, "band_10__2"]], bilinear_interpolate_values="NONE")
            .save(Extract_Multi_Values_to_Points)


        # Process: Copy Features (Copy Features) (management)
        Points10Values_copy =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Points10Values_copy"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
```

```
Approach\Bank Slope Fifth Approach.gdb"):

arcpy.management.CopyFeatures(in_features=Points_withAllvalues,
out_feature_class=Points10Values_copy, config_keyword="",
spatial_grid_1=None, spatial_grid_2=None, spatial_grid_3=None)

        # Process: Table To Table (Table To Table) (conversion)
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Bank5out =
arcpy.conversion.TableToTable(in_rows=Points10Values_copy,
out_path=Bank_Slope_Fifth_Approach_gdb, out_name="Bank5out",
where_clause="", field_mapping="Id \"Id\" true false false 6 Long 0
6,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth Approach.gdb\\Points10Values_copy,Id,-
1,-1;ORIG_FID \"ORIG_FID\" true false false 10 Long 0
10,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,ORIG_FID,-1,-1;band_1_r_1
\"band_1_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_1_r_1,-1,-1;band_2_r_1
\"band_2_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_2_r_1,-1,-1;band_3_r_1
\"band_3_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_3_r_1,-1,-1;band_4_r_1
\"band_4_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_4_r_1,-1,-1;band_5_r_1
\"band_5_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_5_r_1,-1,-1;band_6_r_1
\"band_6_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_6_r_1,-1,-1;band_7_r_1
\"band_7_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_7_r_1,-1,-1;band_8_r_1
\"band_8_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_8_r_1,-1,-1;band_9_r_1
\"band_9_r_1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
Approach.gdb\\Points10Values_copy,band_9_r_1,-1,-1;band_10__1
\"band_10__1\" true false false 13 Float 0
0,First,#,C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope
Fifth Approach\\Bank Slope Fifth
```

```
Approach.gdb\\Points10Values_copy,band_10__1,-1,-1",
config_keyword="")[0]

        # Process: IDW (6) (IDW) (3d)
        Idw_river1_6 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_6"
        with arcpy.EnvManager(extent=Green_LiDar,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues,
z_field="band_6_r_1", out_raster=Idw_river1_6, cell_size="1", power=10,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_6 = arcpy.Raster(Idw_river1_6)

        # Process: IDW (IDW) (3d)
        Idw_river1_1 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_1"
        with arcpy.EnvManager(extent=Green_LiDar,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues,
z_field="band_1_r_1", out_raster=Idw_river1_1, cell_size="1", power=10,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_1 = arcpy.Raster(Idw_river1_1)

        # Process: Raster Calculator (11) (Raster Calculator) (ia)
        Idw_M1_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M1_boundary.tif"
        Raster_Calculator_11_ = Idw_M1_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M1_boundary_tif =  SetNull(Idw_river1_1 <Green_LiDar
,Idw_river1_1)
            Idw_M1_boundary_tif.save(Raster_Calculator_11_)


        # Process: Raster Domain (Raster Domain) (3d)
        Idw_M1_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M1_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M1_boundary_tif,
out_feature_class=Idw_M1_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (2) (IDW) (3d)
        Idw_river1_2 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_2"
        with arcpy.EnvManager(extent=Green_LiDar,
```

```
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues,
z_field="band_2_r_1", out_raster=Idw_river1_2, cell_size="1", power=10,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_2 = arcpy.Raster(Idw_river1_2)

        # Process: Raster Calculator (12) (Raster Calculator) (ia)
        Idw_M2_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M2_boundary.tif"
        Raster_Calculator_12_ = Idw_M2_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M2_boundary_tif =  SetNull(Idw_river1_2 <Green_LiDar
,Idw_river1_2)
            Idw_M2_boundary_tif.save(Raster_Calculator_12_)


        # Process: Raster Domain (2) (Raster Domain) (3d)
        Idw_M2_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M2_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M2_boundary_tif,
out_feature_class=Idw_M2_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (3) (IDW) (3d)
        Idw_river1_3 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_3"
        with arcpy.EnvManager(extent=Green_LiDar,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues,
z_field="band_3_r_1", out_raster=Idw_river1_3, cell_size="1", power=10,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_3 = arcpy.Raster(Idw_river1_3)

        # Process: Raster Calculator (13) (Raster Calculator) (ia)
        Idw_M3_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M3_boundary.tif"
        Raster_Calculator_13_ = Idw_M3_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M3_boundary_tif =  SetNull(Idw_river1_3 <Green_LiDar
,Idw_river1_3)
            Idw_M3_boundary_tif.save(Raster_Calculator_13_)
```

```
        # Process: Raster Domain (3) (Raster Domain) (3d)
        Idw_M3_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M3_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M3_boundary_tif,
out_feature_class=Idw_M3_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (4) (IDW) (3d)
        Idw_river1_4 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_4"
        with arcpy.EnvManager(extent=Green_LiDar,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues,
z_field="band_4_r_1", out_raster=Idw_river1_4, cell_size="1", power=10,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_4 = arcpy.Raster(Idw_river1_4)

        # Process: Raster Calculator (14) (Raster Calculator) (ia)
        Idw_M4_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M4_boundary.tif"
        Raster_Calculator_14_ = Idw_M4_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M4_boundary_tif =  SetNull(Idw_river1_4 <Green_LiDar
,Idw_river1_4)
            Idw_M4_boundary_tif.save(Raster_Calculator_14_)


        # Process: Raster Domain (4) (Raster Domain) (3d)
        Idw_M4_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M4_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M4_boundary_tif,
out_feature_class=Idw_M4_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (5) (IDW) (3d)
        Idw_river1_5 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_5"
        with arcpy.EnvManager(extent=Green_LiDar,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues,
```

```
z_field="band_5_r_1", out_raster=Idw_river1_5, cell_size="1", power=10,
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_5 = arcpy.Raster(Idw_river1_5)

        # Process: Raster Calculator (15) (Raster Calculator) (ia)
        Idw_M5_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M5_boundary.tif"
        Raster_Calculator_15_ = Idw_M5_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M5_boundary_tif =  SetNull(Idw_river1_5 <Green_LiDar
,Idw_river1_5)
            Idw_M5_boundary_tif.save(Raster_Calculator_15_)


        # Process: Raster Domain (5) (Raster Domain) (3d)
        Idw_M5_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M5_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M5_boundary_tif,
out_feature_class=Idw_M5_Domain_shp, out_geometry_type="POLYGON")

        # Process: Raster Calculator (16) (Raster Calculator) (ia)
        Idw_M6_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M6_boundary.tif"
        Raster_Calculator_16_ = Idw_M6_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M6_boundary_tif =  SetNull(Idw_river1_5 <Green_LiDar
,Idw_river1_5)
            Idw_M6_boundary_tif.save(Raster_Calculator_16_)


        # Process: Raster Domain (6) (Raster Domain) (3d)
        Idw_M6_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M6_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M6_boundary_tif,
out_feature_class=Idw_M6_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (7) (IDW) (3d)
        Idw_river1_7 =
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_7"
        with arcpy.EnvManager(extent=Green_LiDar,
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope
Fifth Approach\Bank Slope Fifth Approach.gdb",
```

```
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues, z_field="band_7_r_1", out_raster=Idw_river1_7, cell_size="1", power=10, search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_7 = arcpy.Raster(Idw_river1_7)

        # Process: Raster Calculator (17) (Raster Calculator) (ia)
        Idw_M7_boundary_tif = "E:\\GIS Automation\\Bank Slope_5\\Site_1\\IDW\\Idw_M7_boundary.tif"
        Raster_Calculator_17_ = Idw_M7_boundary_tif
        with arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M7_boundary_tif =  SetNull(Idw_river1_7 <Green_LiDar ,Idw_river1_7)
            Idw_M7_boundary_tif.save(Raster_Calculator_17_)


        # Process: Raster Domain (7) (Raster Domain) (3d)
        Idw_M7_Domain_shp = "E:\\GIS Automation\\Bank Slope_5\\Site_1\\Domain\\Idw_M7_Domain.shp"
        with arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M7_boundary_tif, out_feature_class=Idw_M7_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (8) (IDW) (3d)
        Idw_river1_8 = "C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_8"
        with arcpy.EnvManager(extent=Green_LiDar, scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues, z_field="band_8_r_1", out_raster=Idw_river1_8, cell_size="1", power=10, search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_8 = arcpy.Raster(Idw_river1_8)

        # Process: Raster Calculator (18) (Raster Calculator) (ia)
        Idw_M8_boundary_tif = "E:\\GIS Automation\\Bank Slope_5\\Site_1\\IDW\\Idw_M8_boundary.tif"
        Raster_Calculator_18_ = Idw_M8_boundary_tif
        with arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M8_boundary_tif =  SetNull(Idw_river1_8 <Green_LiDar ,Idw_river1_8)
            Idw_M8_boundary_tif.save(Raster_Calculator_18_)


        # Process: Raster Domain (8) (Raster Domain) (3d)
```

```
        Idw_M8_Domain_shp = "E:\\GIS Automation\\Bank 
Slope_5\\Site_1\\Domain\\Idw_M8_Domain.shp"
        with 
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth 
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M8_boundary_tif, 
out_feature_class=Idw_M8_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (9) (IDW) (3d)
        Idw_river1_9 = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth 
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_9"
        with arcpy.EnvManager(extent=Green_LiDar, 
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope 
Fifth Approach\Bank Slope Fifth Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth 
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues, 
z_field="band_9_r_1", out_raster=Idw_river1_9, cell_size="1", power=10, 
search_radius="VARIABLE 12", in_barrier_polyline_features="")
            Idw_river1_9 = arcpy.Raster(Idw_river1_9)

        # Process: Raster Calculator (19) (Raster Calculator) (ia)
        Idw_M9_boundary_tif = "E:\\GIS Automation\\Bank 
Slope_5\\Site_1\\IDW\\Idw_M9_boundary.tif"
        Raster_Calculator_19_ = Idw_M9_boundary_tif
        with 
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth 
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M9_boundary_tif =  SetNull(Idw_river1_9 <Green_LiDar 
,Idw_river1_9)
            Idw_M9_boundary_tif.save(Raster_Calculator_19_)


        # Process: Raster Domain (9) (Raster Domain) (3d)
        Idw_M9_Domain_shp = "E:\\GIS Automation\\Bank 
Slope_5\\Site_1\\Domain\\Idw_M9_Domain.shp"
        with 
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth 
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M9_boundary_tif, 
out_feature_class=Idw_M9_Domain_shp, out_geometry_type="POLYGON")

        # Process: IDW (10) (IDW) (3d)
        Idw_river1_10 = 
"C:\\Users\\moawadal\\Documents\\ArcGIS\\Projects\\Bank Slope Fifth 
Approach\\Bank Slope Fifth Approach.gdb\\Idw_river1_10"
        with arcpy.EnvManager(extent=Green_LiDar, 
scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope 
Fifth Approach\Bank Slope Fifth Approach.gdb", 
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth 
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.Idw(in_point_features=Points_withAllvalues, 
z_field="band_10__1", out_raster=Idw_river1_10, cell_size="1", power=10, 
search_radius="VARIABLE 12", in_barrier_polyline_features="")
```

```
            Idw_river1_10 = arcpy.Raster(Idw_river1_10)

        # Process: Raster Calculator (20) (Raster Calculator) (ia)
        Idw_M10_boundary_tif = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\IDW\\Idw_M10_boundary.tif"
        Raster_Calculator_20_ = Idw_M10_boundary_tif
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            Idw_M10_boundary_tif =  SetNull(Idw_river1_10 <Green_LiDar
,Idw_river1_10)
            Idw_M10_boundary_tif.save(Raster_Calculator_20_)


        # Process: Raster Domain (10) (Raster Domain) (3d)
        Idw_M10_Domain_shp = "E:\\GIS Automation\\Bank
Slope_5\\Site_1\\Domain\\Idw_M10_Domain.shp"
        with
arcpy.EnvManager(scratchWorkspace=r"C:\Users\moawadal\Documents\ArcGIS\Pr
ojects\Bank Slope Fifth Approach\Bank Slope Fifth Approach.gdb",
workspace=r"C:\Users\moawadal\Documents\ArcGIS\Projects\Bank Slope Fifth
Approach\Bank Slope Fifth Approach.gdb"):
            arcpy.ddd.RasterDomain(in_raster=Idw_M10_boundary_tif,
out_feature_class=Idw_M10_Domain_shp, out_geometry_type="POLYGON")

if __name__ == '__main__':
    Model1()
```

Postprocessing code for the bank's slope in R

```
dir_1 <- "E://GIS Automation//Bank Slope_5//Site_"
dir_wse <- "E://GIS Automation//Bank Slope_5//Site_"
dir_wse_1 <- "E://GIS Automation//Bank Slope_3//Site_"
dir<- "E://GIS Automation//LiDAR Error Transects//Site_"
out_dir <- "E://GIS Automation//Comparisons//Bank Slope_4//Total Site_"
dir_2 <- "E://GIS Automation//Missed Volume Transects//Sites//output//Missed Volume
Transect_"
s <- "//output//BankSideSlopes6_T_"
x <- "//output//LiDAR Error Transect_ClipDS_T_"
y <- ".csv"
library(ggplot2)
library(matrixStats)
library(corrplot)
library(broom)
library(gridExtra)
library(matrixStats)
library(tidyverse)
for (i in 3:3){
  print(i)
  q <- list()
  for(j in 1:7){
    data_dir_B <- paste(dir_1,i,s,i, "_",j,y, sep = "")
    data_dir_Tr <- paste(dir,i,x,i,y, sep = "")
    data_dir_M <- paste(dir_2,i,y, sep = "")
    m_folder <- paste0(dir_wse,i,"//Output//Intervals_",i,"_M.csv")
    M_mat <-  read.csv(m_folder,header = T, sep = ",")
    output <- paste(out_dir, i,".csv", sep = "")
    m <- read.csv(data_dir_B,header = T, sep = ",")
    m <- arrange(m,m$V1)
    l <- read.csv(data_dir_Tr,header = T, sep = ",")
    n <- read.csv(data_dir_M,header = T, sep = ",")
    data_all <- cbind(m,n,l)
    z_matrix <- data_all[,c(2:19)]

    wse_folder <- paste0(dir_wse_1,i,"//WSE//WSE_Transect_G1_",i,".csv")
    wse_folderR <- paste0(dir_wse_1,i,"//WSE_R//WSE_Transect_R_",i,"7.csv")
    wse <- read.csv(wse_folder,header = T, sep = ",")
    wse_r <- read.csv(wse_folderR,header = T, sep = ",")
    nrow1 <- nrow(m)
    result_matrix <- matrix(NA,nrow = nrow1,ncol = 9)
    for(k in 1:nrow(m)) {
      wse1 <- wse$RASTERVALU[k]
      wse2 <- wse_r$RASTERVALU[k]
      m_values <- M_mat[k,c(16:25)]-wse1
      m_values2 <- M_mat[k,c(16:25)]-wse2

      for(k1 in 1:9){
        if((m_values[k1] < 0) && (m_values[k1+1] > 0)) {

          e <- k1

        }
```

```
      }
      for (k2 in 1:9) {
        if((m_values2[k2] < 0) && (m_values2[k2+1] > 0)) {

          b <- k2

        }
      }
       if (e == b | b < e){
         result_matrix[k,1] <-rowMeans(z_matrix[k, c((2*e-1), (2*b-1))]) # Right Mean
         result_matrix[k,2] <-rowSds(as.matrix(z_matrix[k,c((2*e-1), (2*b-1))]))# Right SD
         result_matrix[k,3] <-rowMeans(z_matrix[k,c((2*e), (2*b))]) # Left Mean
         result_matrix[k,4] <-rowSds(as.matrix(z_matrix[k,c((2*e), (2*b))])) # Left SD
         result_matrix[k,5] <- rowMeans(z_matrix[k,c((2*e-1):(2*b))]) # Total Mean
         result_matrix[k,6] <-rowSds(as.matrix(z_matrix[k,c((2*e-1):(2*b))])) # Total SD
         result_matrix[k,7] <- sum(data_all$Vol_Diff[1:k])


       }
        else {
          result_matrix[k,1] <-rowMeans(z_matrix[k, seq((2*e-1), (2*b-1),2)]) # Right Mean
          result_matrix[k,2] <-rowSds(as.matrix(z_matrix[k,seq((2*e-1), (2*b-1), 2)]))# Right SD
          result_matrix[k,3] <-rowMeans(z_matrix[k,seq((2*e), (2*b), 2)]) # Left Mean
          result_matrix[k,4] <-rowSds(as.matrix(z_matrix[k,seq((2*e), (2*b), 2)])) # Left SD
          result_matrix[k,5] <- rowMeans(z_matrix[k,c((2*e-1):(2*b))]) # Total Mean
          result_matrix[k,6] <-rowSds(as.matrix(z_matrix[k,c((2*e-1):(2*b))])) # Total SD
          result_matrix[k,7] <- sum(data_all$Vol_Diff[1:k])
        }
        }




    result_matrix[,8] <- l[,j+1]
    result_matrix[,9] <- m$V1


    final_mat <- result_matrix
    colnames(final_mat) <- c("ZR Mean", "ZR SD", "ZL Mean", "ZL SD", "Z Mean", "Z SD",
"Missed Vol", "LiDAR Er", "TrNo")

    dd1 <- data.frame(final_mat)
    dd1 <- na.omit(dd1)
    dd1 <- transform(dd1, Zmean_deg = atan_d(1/(Z.Mean)), Zsd_deg= atan_d(1/(Z.SD)))
    dd2 <- dd1[,-9]
    out_dir_1 <- paste0("E://GIS Automation//Comparisons//Bank Slope_7//Site_", i)
    title_1 <- c("Q Mean Flood", "Q 10-years", "Q 20-years", "Q 50-years", "Q 100-years", "Q 200-
years", "Q 500-years")
    plot_name_1 <- paste0("Comparison_Site_", i, "_3.jpg")
    setwd(out_dir_1)
    reg_1 <- lm(dd1$LiDAR.Er~dd1$Z.Mean+dd1$Z.SD - 1)
    R_sq <- summary(reg_1)$r.squared
    R_sq1 <- format(round(R_sq,digits = 2), nsmall = 2)
    print("Enter the list")
    write.csv( tidy( reg_1 ) , paste0("reg_coeff_", j ,".csv") )
```

```
    write.csv( glance( reg_1 ) , paste0("reg_table_", j ,".csv"))
    tiff(paste0("Cor_",j,"_",plot_name_1), units = "in", width = 8, height = 6, res = 400)
    corrplot::corrplot(cor(dd2),method = "color", addCoef.col = "white")
    dev.off()
    q[[j]] <- ggplot(dd1) + geom_point(aes(y=LiDAR.Er, x=Zmean_deg, colour= Zsd_deg)) +
      scale_colour_gradientn(colours=rainbow(4)) +
      labs(x="Bank Slope (Degree)", y="Inundation Error (%)", color="SD (Degree)") +
      ggtitle(title_1[j]) +
      theme(axis.text=element_text(size=18, colour = "black"),axis.title=element_text(size=18),
legend.title = element_text(color = "black", size = 18),
            legend.text = element_text(color = "black", size = 14, face = "italic"),
            panel.background = element_blank(), axis.line = element_line(colour = "black", size = 0.8),
legend.position = c(0.8, 0.5), plot.title = element_text(color = "black", size = 20, face = "bold",
hjust = 0.5))
    print(paste0("save in list Plot No_", j))
     }
  print("finished the list")
  library(gridExtra)

  png(filename = plot_name_1 , width = 1280, height = 960, units = "px")
  do.call(grid.arrange, q)
  dev.off()
  print("Plot saved successfully!")
  }
```

Bank’s slope approaches using ***Points***

R Language code

```
library(matrixStats)
library(tidyverse)
for (i in 1:1) {
  for (m in 1:1) {
    dir_Tr <- paste0("E://GIS Automation//LiDAR Error Transects//Site_", i, "//output//LiDAR Error Transect_ClipDS_T_", m, ".csv")
    data_Tr <- read.csv(dir_Tr, header = T, sep = ",")
    dir_R <- paste0("E://GIS Automation//Bank Slope Points_1//Site_", i, "//Points//Table_Red_", m, "_.csv")#Directory of the Red LiDAR Scenario
    data_R <- read.csv(dir_R, header = T, sep = ",") #Red LiDAR scenario points
    dir_G <- paste0("E://GIS Automation//Bank Slope Points_1//Site_", i, "//Points//Table_Green_", m, "_.csv")#Directory of the Green LiDAR Scenario
    data_G <- read.csv(dir_G, header = T, sep = ",") #Green LiDAR Scenario point
    dir_Ter <- paste0("E://GIS Automation//Bank Slope Points_1//Site_" , i ,"//Points//Table_Terrain_P_", i , "_.csv")
    data1 <- read.csv(dir_Ter, header = T, sep = ",") #Points along the river line (has x y z and the number of the transect)
    ntransects <- nrow(data1) #number of row (transect)
    result_matrix <- matrix(NA,nrow = ntransects,ncol = 8)
    zmatrix_total <- matrix(NA, nrow = ntransects, ncol = 20)
    for (j in 1: ntransects) {
      dir_Table_P <- paste0("E://GIS Automation//Bank Slope Points_1//Site_", i, "//Points_in_table//Table_P_", j, ".csv")#Directory of the table of the points
      data3 <- read.csv(dir_Table_P, header = T, sep = ",") #read directory
      data2 <-  arrange(data3,Cop_P_ID)
      npoints <- nrow(data2)
      centriod_x <- data1[j,7] # 7 is the poistion of the x coordinate
      centriod_y <- data1[j, 8] # 8 is the position of y coordinate
      centriod_z <- data1[j, 6] # 6 is the position of z coordinate
      Red_Elev <- data_R[j,6] # The WSE of the Red LiDAR Scenario
      Green_Elev <- data_G[j,6] # The WSE of the Green LiDAR Scenario
      dist_xy <- sqrt((data2[,8] - centriod_x)^2 + (data2[,9] - centriod_y)^2)
      assumed_centriod <- data2[(round((npoints)/2, digits = 0)),]
      i1 <- ceiling((npoints/2))
      e <- NA
      ee <- NA
      for (k in i1 : (npoints-1)) { #Right Bank Mostly
        if((data2[k,8] < Red_Elev) && (data2[k+1,8] > Red_Elev)) {

          e <- k+1

          break

        }
      }
      if(is.na(e)==FALSE) {
        for (q in i1 : 2) { #Left Bank Mostly
          if ((data2[q,8] < Red_Elev) && (data2[q-1,8] > Red_Elev)) {
```

```
        ee <- q - 1

        break
      }

    }
  }

  dz <- (Red_Elev - centriod_z)/10
 z1 <- centriod_z + dz
 i2L <- i1
 i2R <- i1
 ee <- ifelse(is.na(e)==TRUE,NA,ee)
  for (w in 1 : 10) {
    if(is.na(ee)==FALSE) {
      x0L <- data2[i2L,9]
      y0L <- data2[i2L,10]
      indL <- i2L-which((data2[(i2L : ee),8]-z1)>0)[1]
      x1L <-  data2[indL,9]
      y1L <-  data2[indL,10]

      dis <- sqrt((x0L - x1L)^2 + (y0L - y1L)^2)
      zmatrix_total[j,w] <- dis/dz

      x0R <- data2[i2R,9]
      y0R <- data2[i2R,10]
      indR <- i2R+which((data2[(i2R:e),8]-z1)>0)[1]
      x1R <-  data2[indR,9]
      y1R <-  data2[indR,10]
      dis <- sqrt((x0R - x1R)^2 + (y0R - y1R)^2)
      zmatrix_total[j,w+10]<- dis/dz


      z1 = z1 + dz
      i2L <- indL
      i2R <- indR
    }else{
      zmatrix_total[j,w] <- NA
      zmatrix_total[j,w+10]<- NA
    }

  }






}
result_matrix[,1] <- rowMeans(zmatrix_total[,c(12:20)]) # Right Mean
result_matrix[,2] <- rowSds(as.matrix(zmatrix_total[,c(12:20)]))# Right SD
result_matrix[,3] <- rowMeans(zmatrix_total[,c(2:10)]) # Left Mean
result_matrix[,4] <- rowSds(as.matrix(zmatrix_total[,c(2:10)])) # Left SD
result_matrix[,5] <- rowMeans(zmatrix_total[,-c(1,11)]) # Total Mean
result_matrix[,6] <- rowSds(zmatrix_total[,-c(1,11)]) # Total SD
```

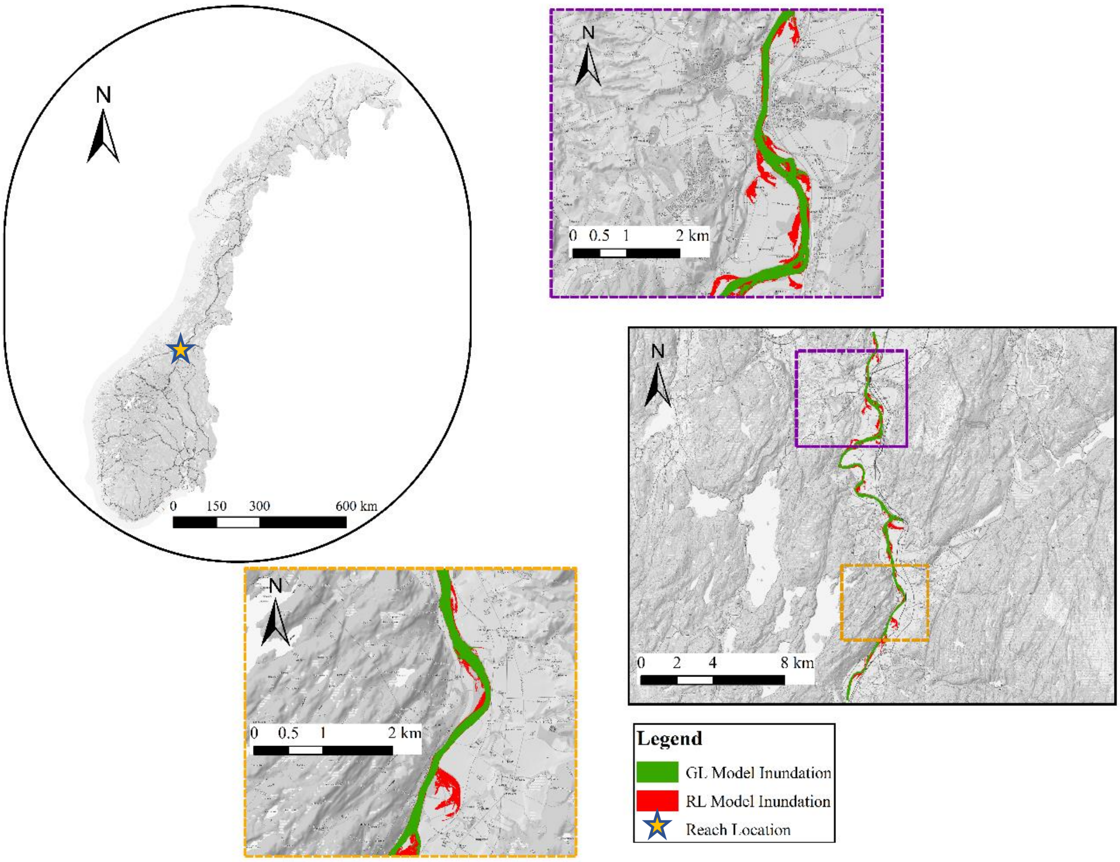
```
    result_matrix[,7] <- data_Tr[,m+1] # LiDAR Error
    result_matrix[,8] <- data1[,1]

  }
}

final_mat <- result_matrix
colnames(final_mat) <- c("ZR Mean", "ZR SD", "ZL Mean", "ZL SD", "Z Mean", "Z SD",
"LiDAR Er", "TrNo")
dd1 <- data.frame(final_mat)
ggplot(dd1, aes(y=LiDAR.Er, x=Z.Mean, colour = Z.SD)) + geom_point(size=2.5) +
  scale_colour_gradientn(colours=rainbow(4)) +
  geom_text(label= dd1$TrNo, size= 4, nudge_x = 0, nudge_y = 5) +
  theme_classic()
```

# Appendix B: Results

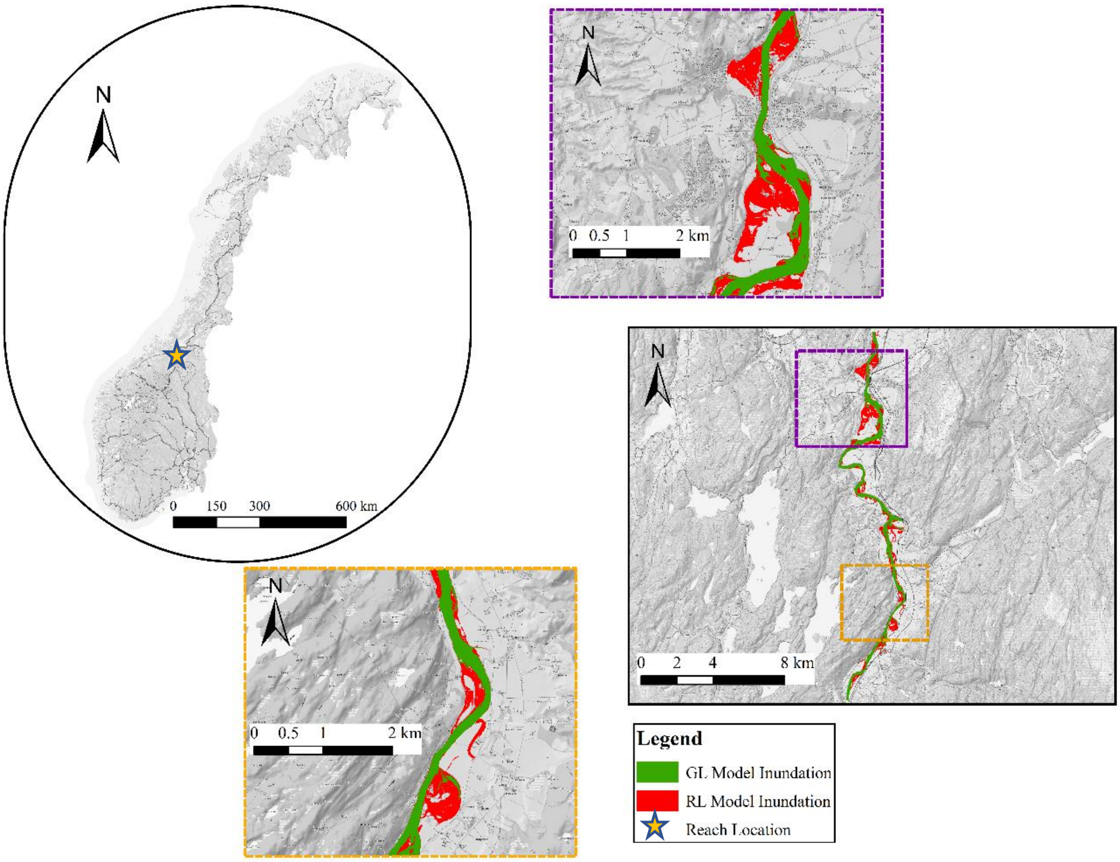
Flood inundation maps

a. Gaula Site

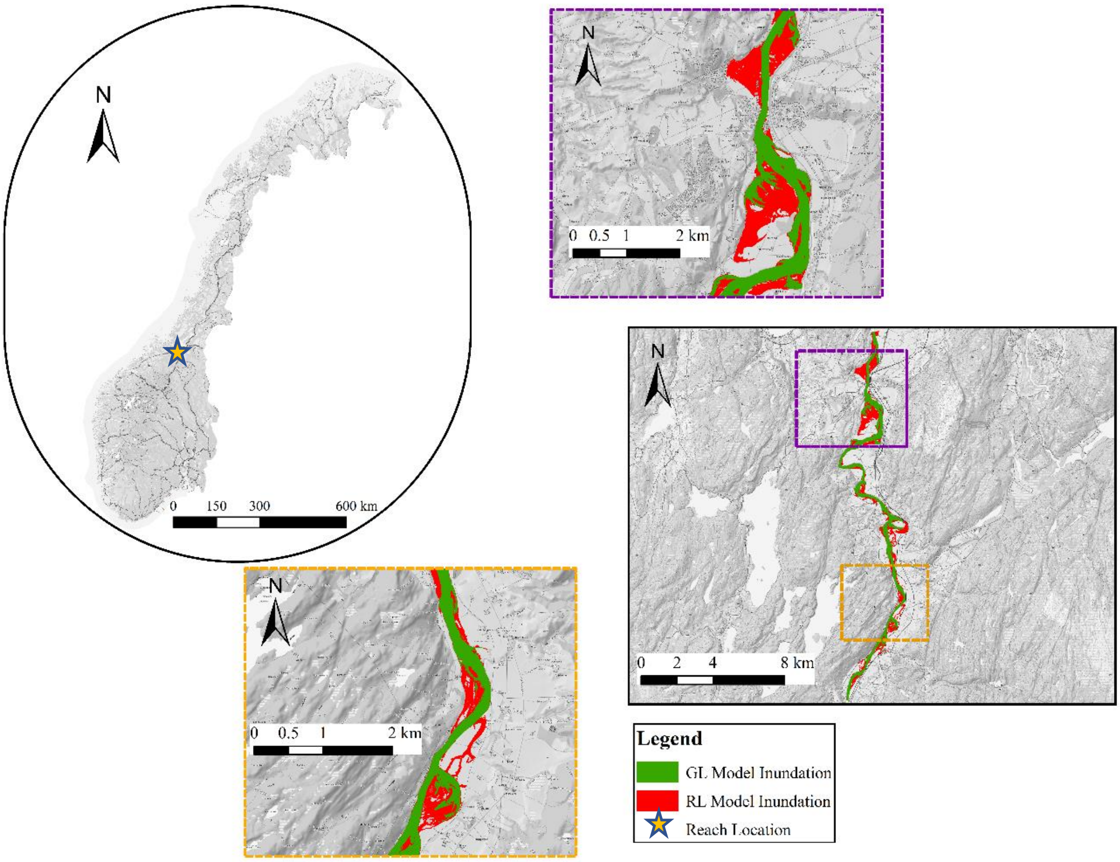
1. Mean flood Scenario inundations



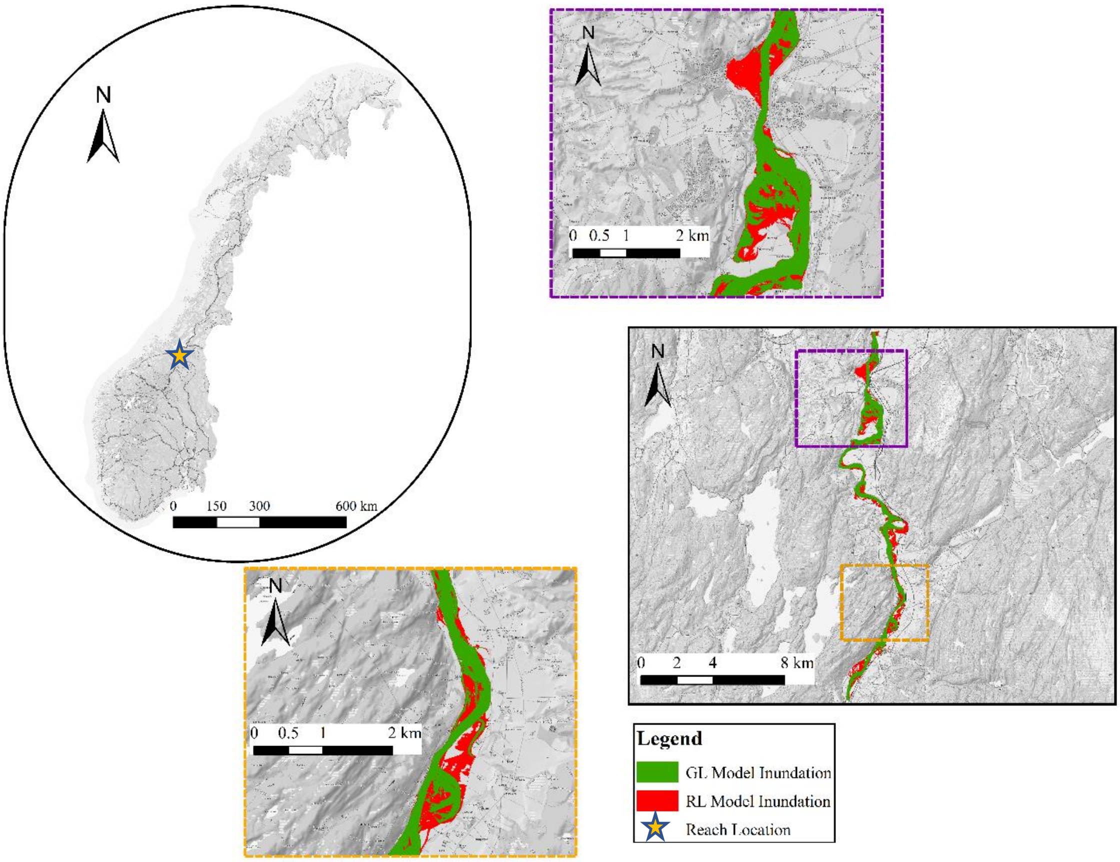
2. 10-years flood Scenario inundations



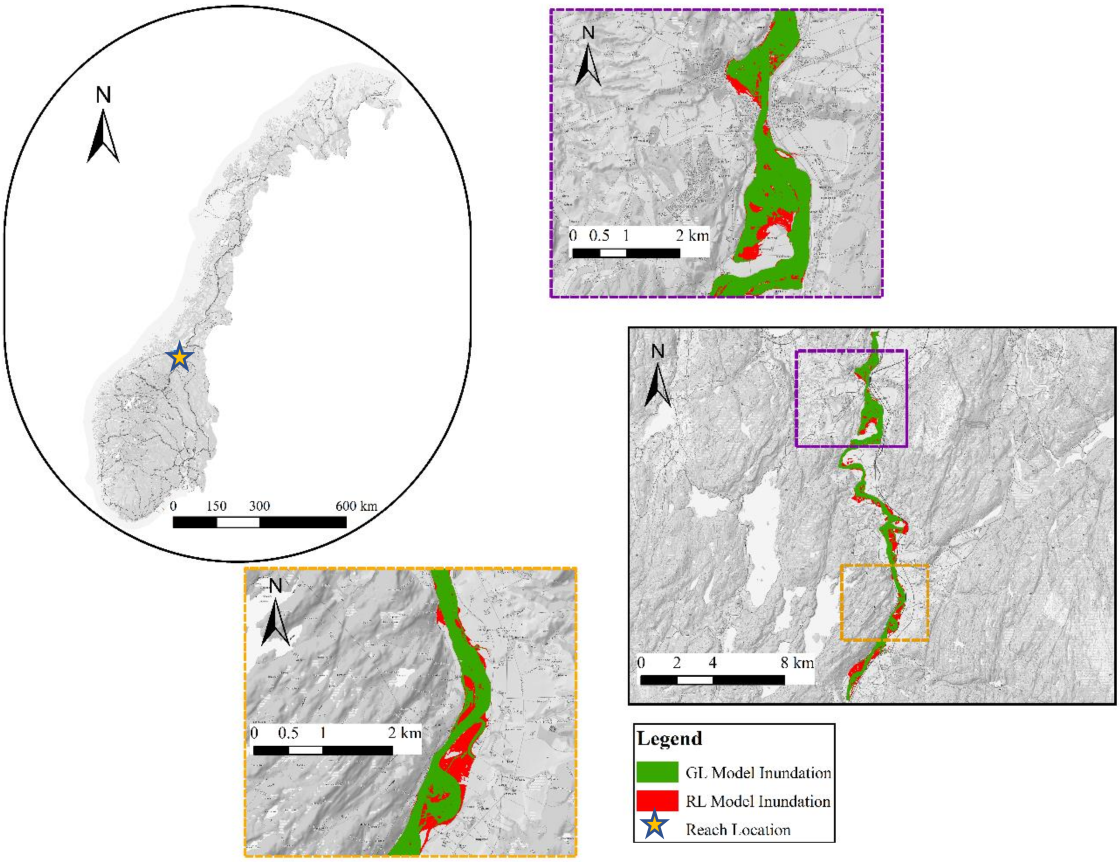
3. 20-years flood Scenario inundations



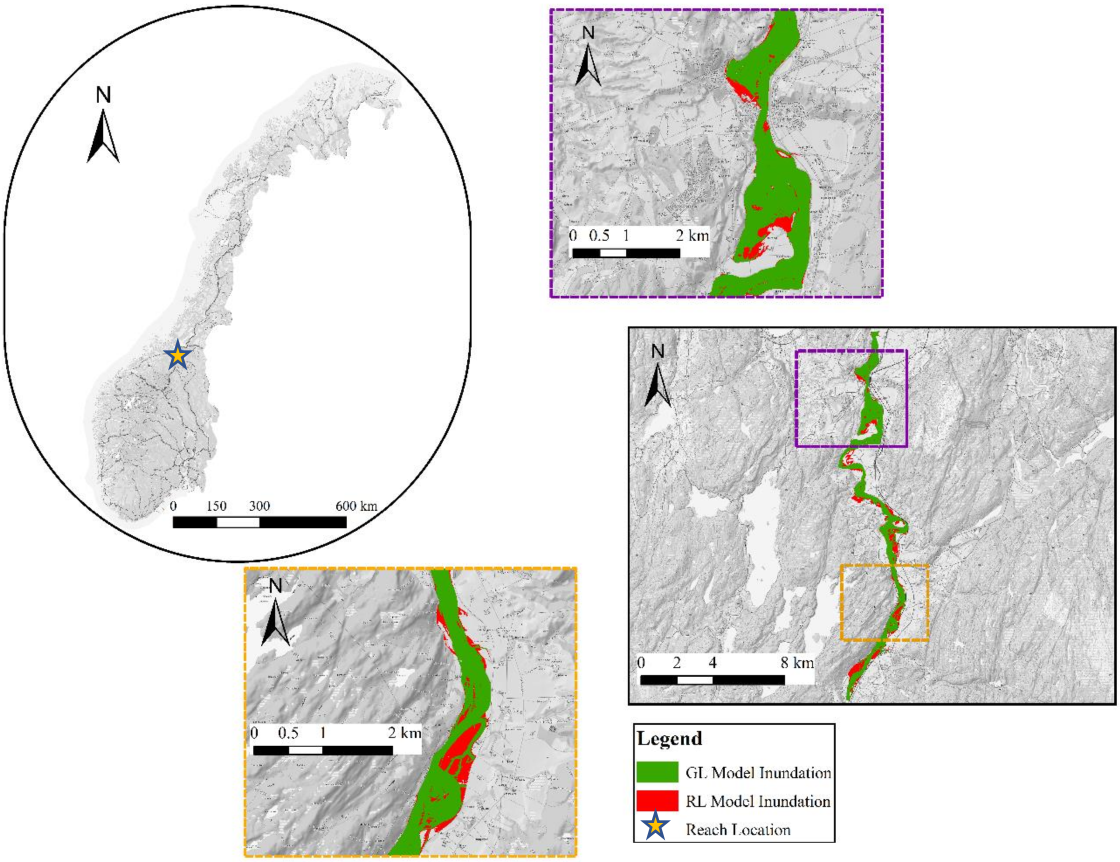
4. 50-years flood Scenario inundations



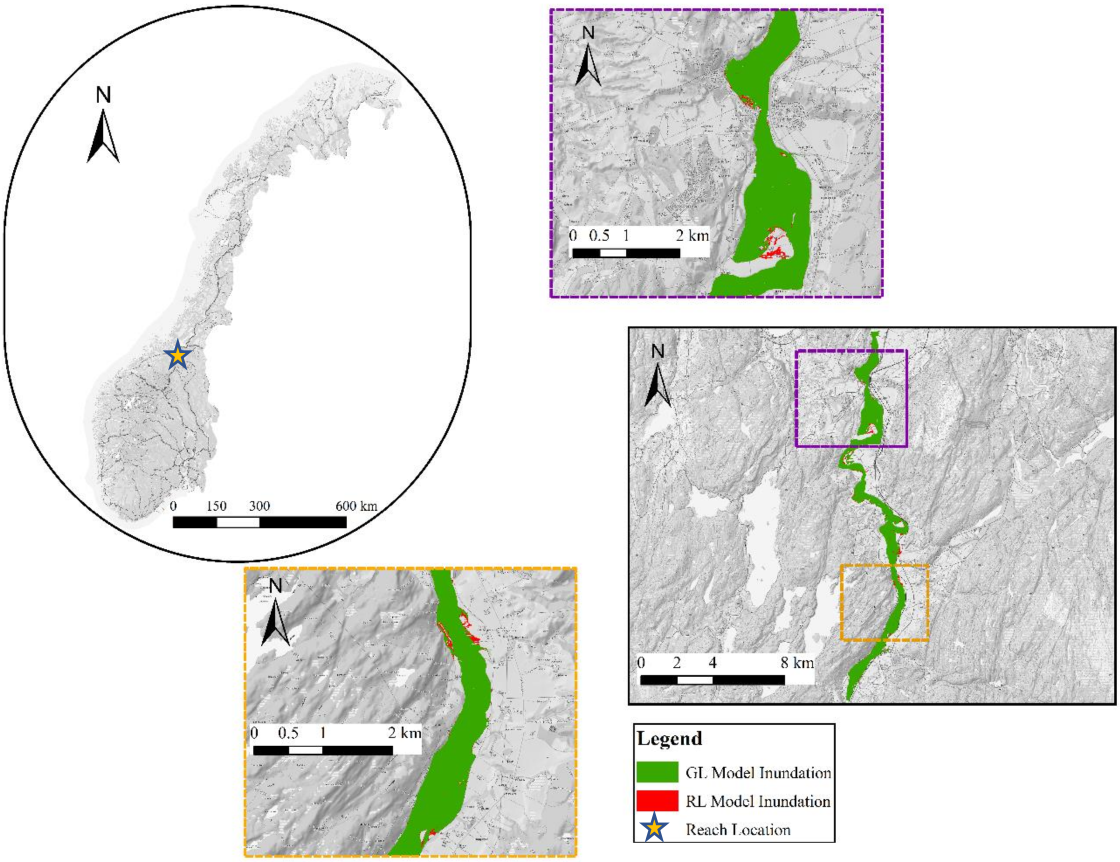
5. 100-years flood Scenario inundations



6. 200-years flood Scenario inundations

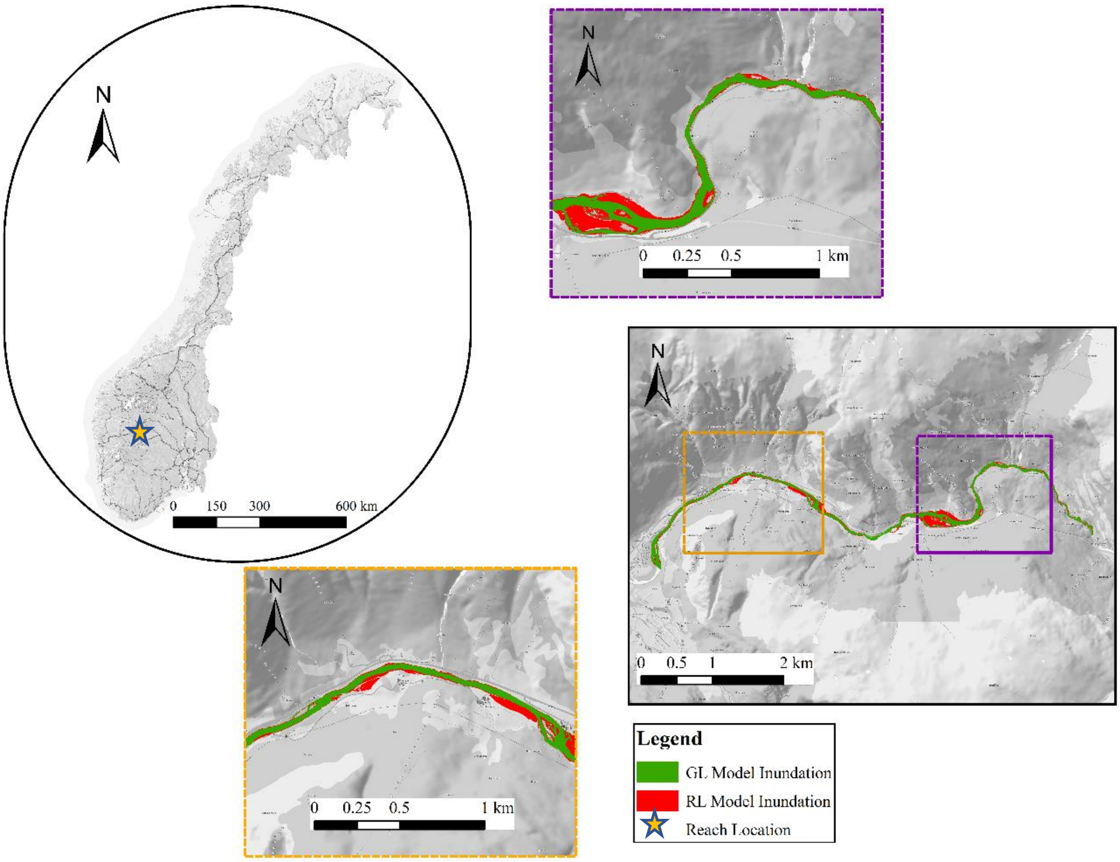


7. 500-years flood Scenario inundations

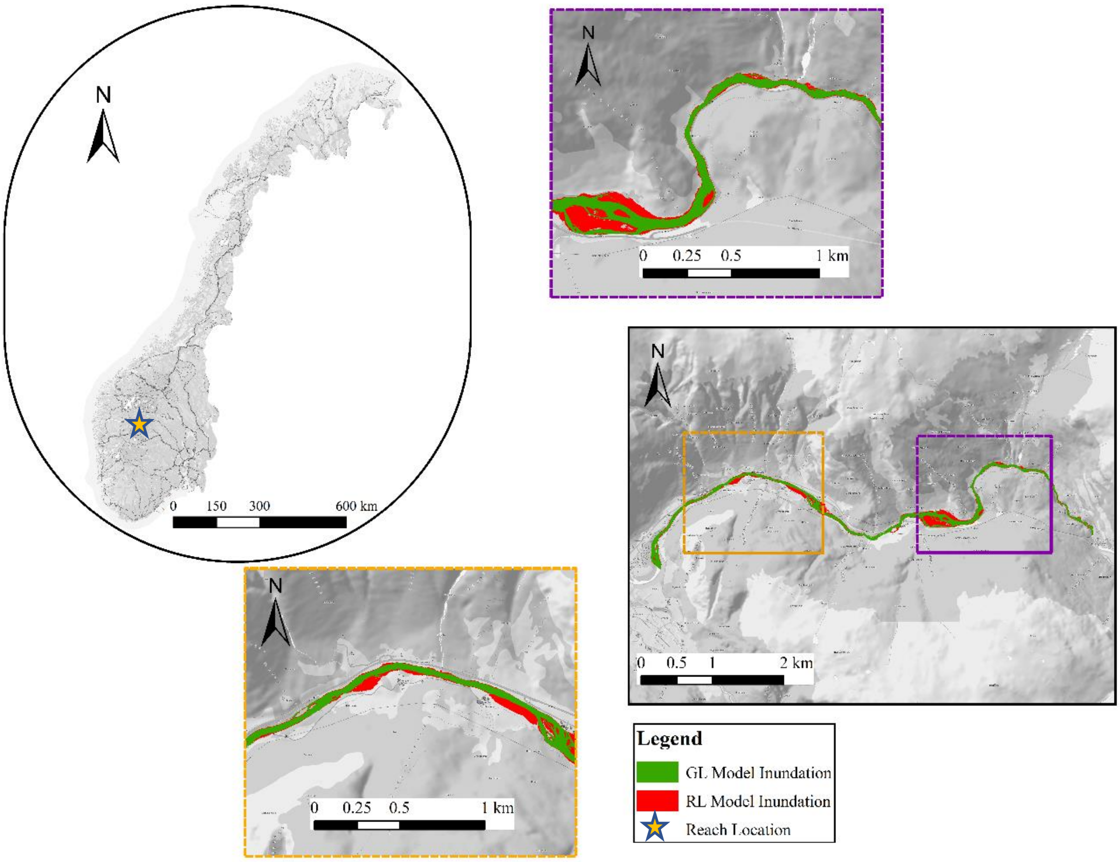


b. Upper Lærdal site

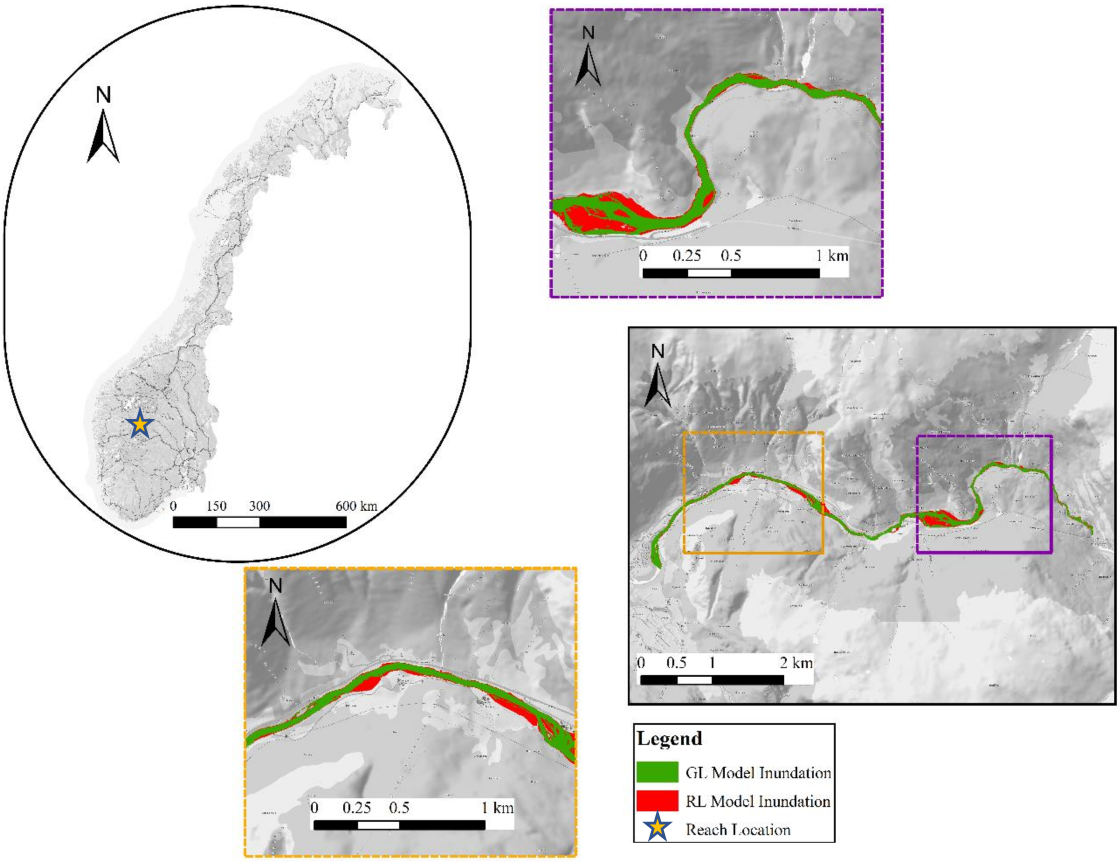
1. Mean flood scenario inundations



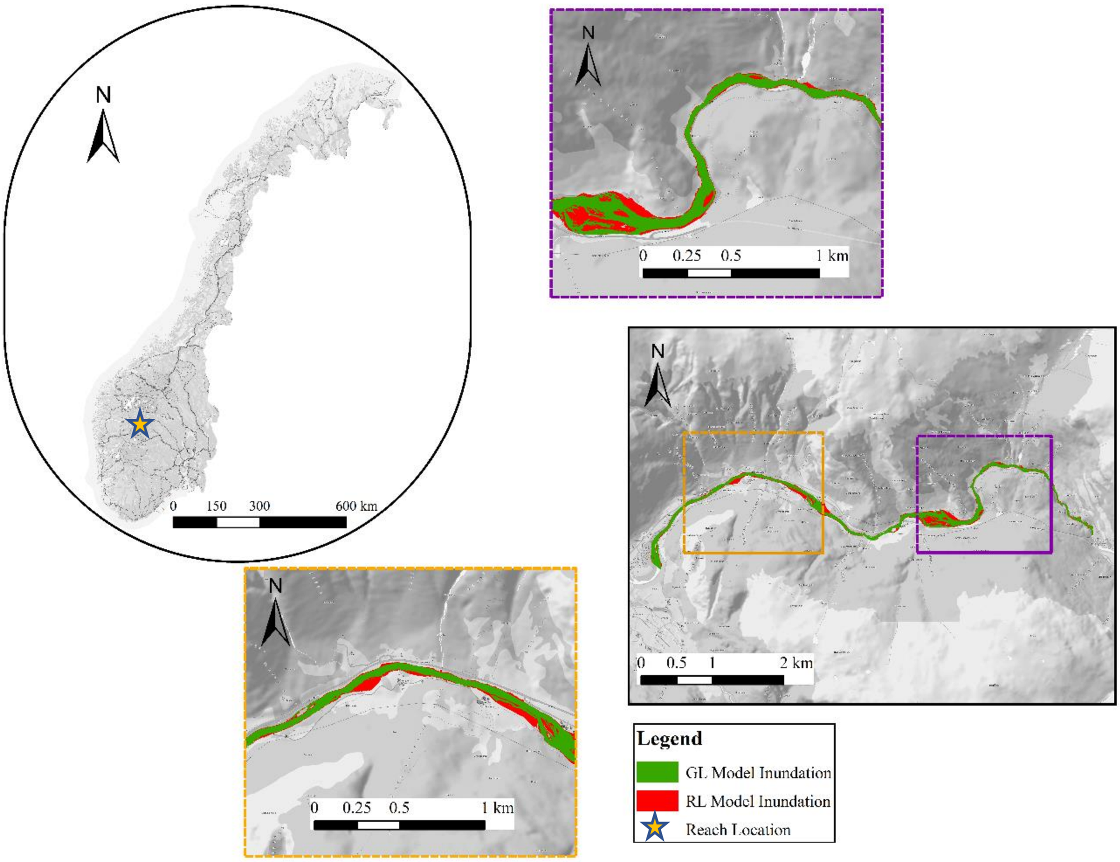
2. 10-years flood scenario inundations



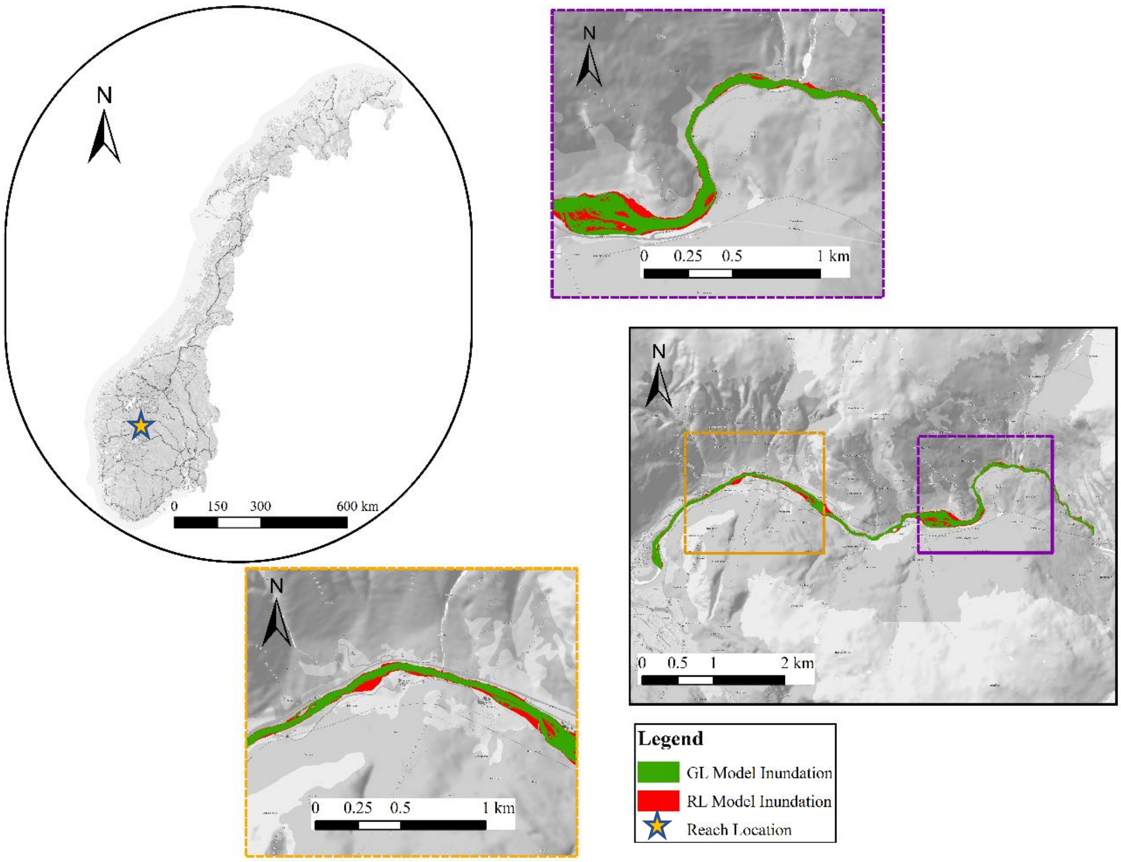
3. 20-years flood scenario inundations



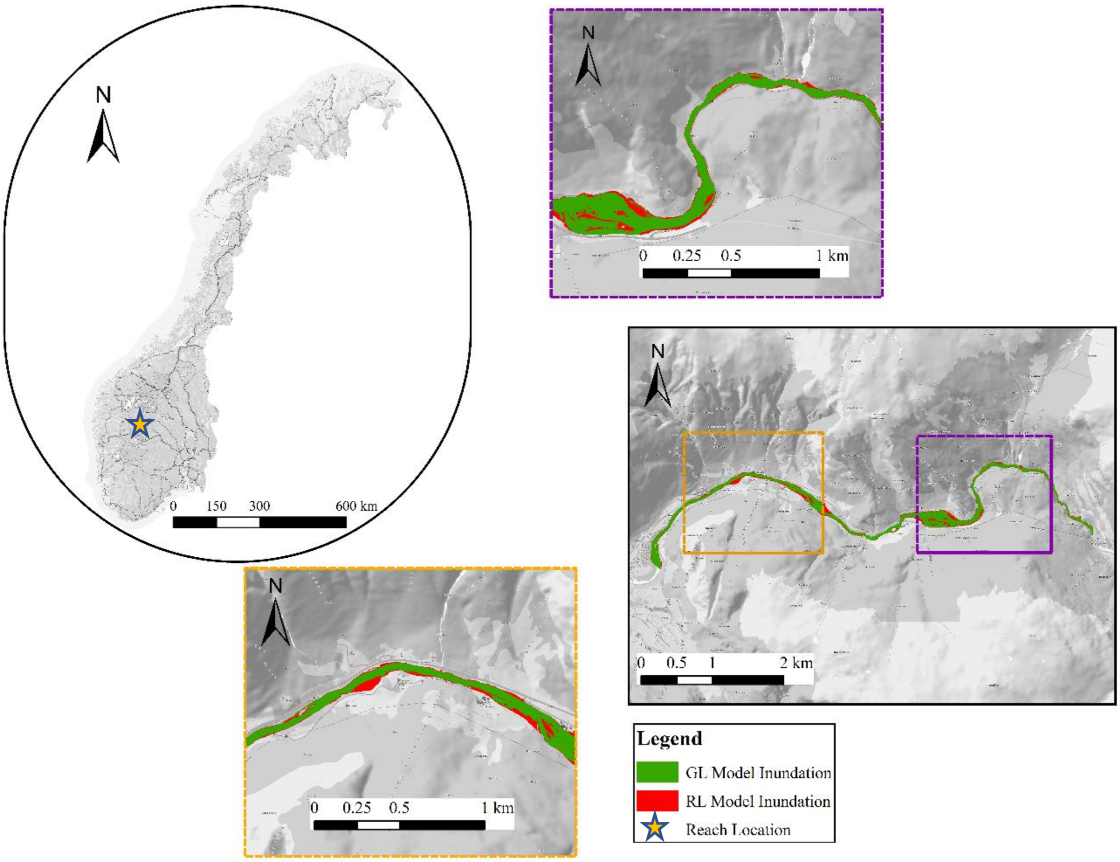
4. 50-years flood scenario inundations



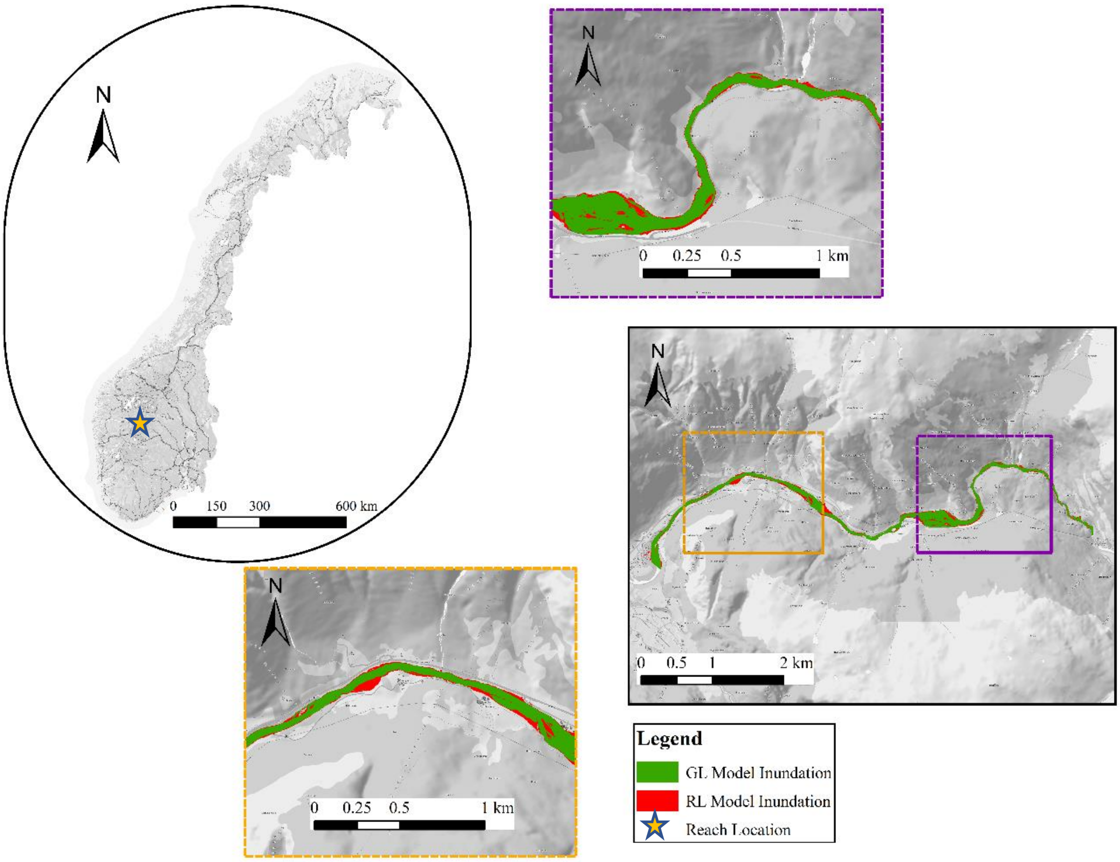
5. 100-years flood scenario inundations



6. 200-years flood scenario inundations



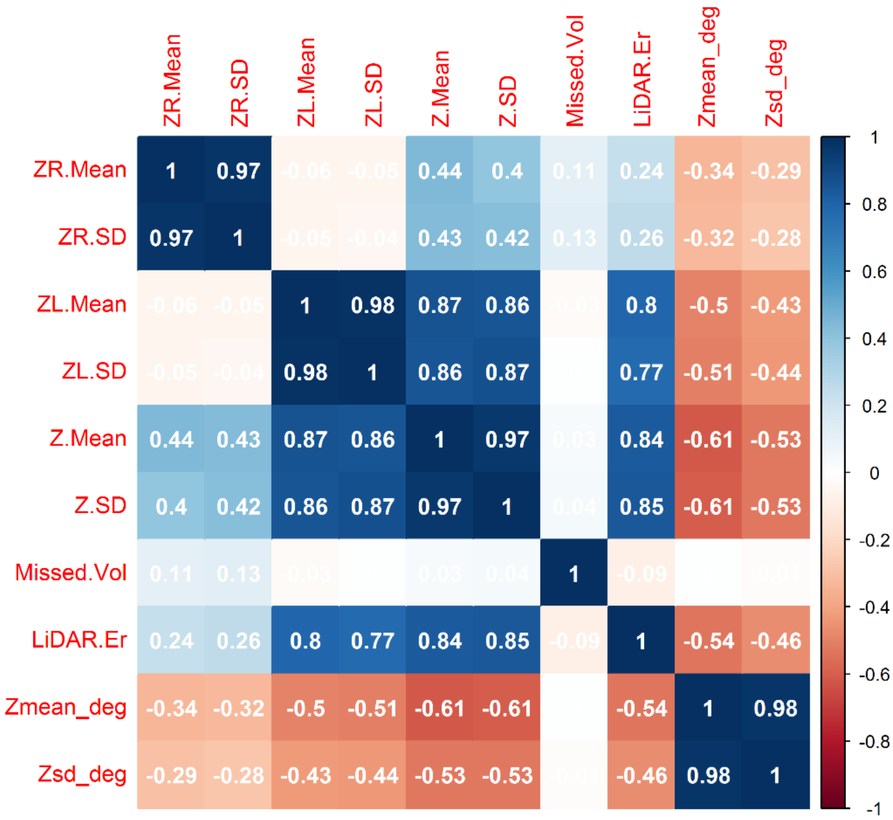
7. 500-years flood scenario inundations



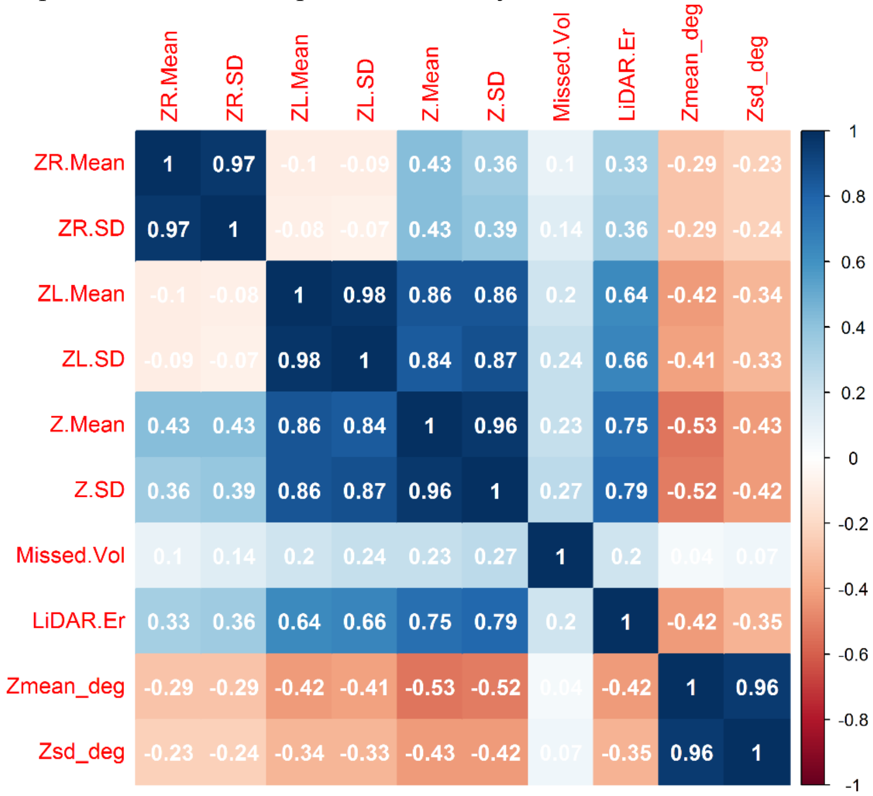
Correlation's relationships

a) Gaula site

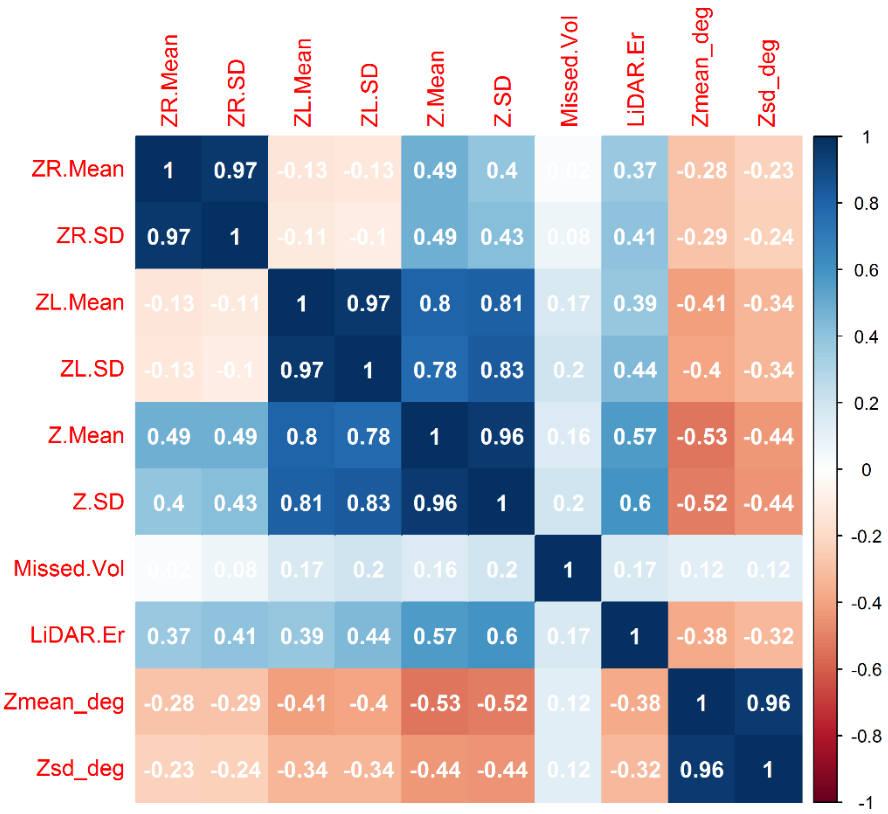
1. Correlation plot for the bank's slope to level of Mean flood scenario



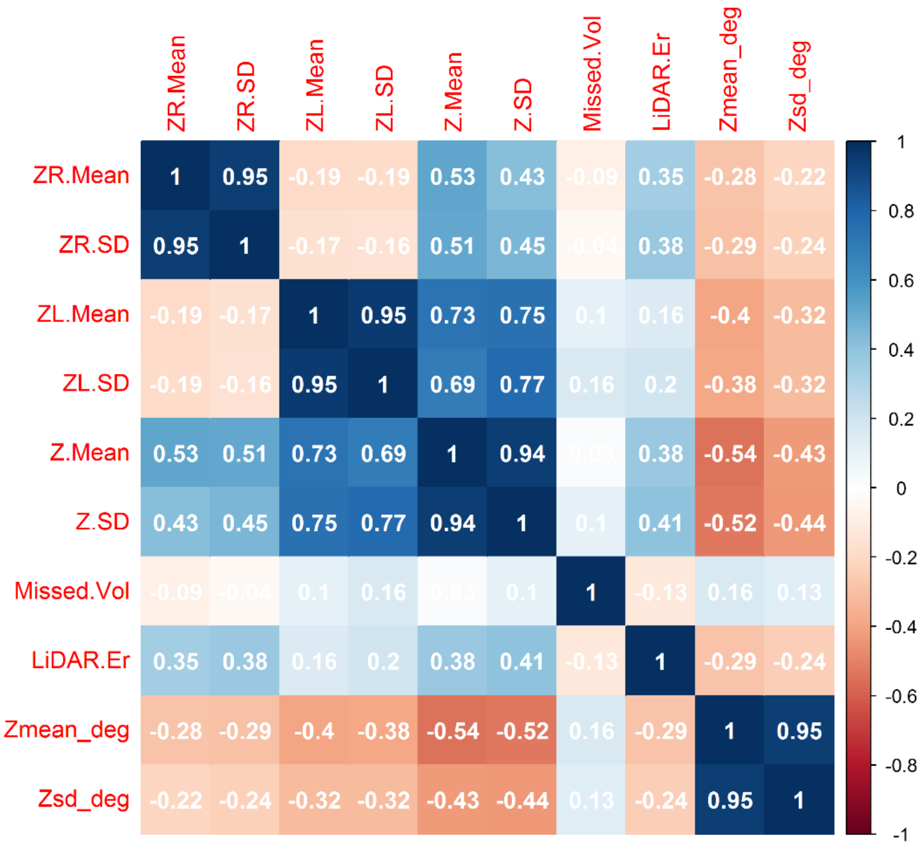
2. Correlation plot for the bank's slope to level of 10-years flood scenario.



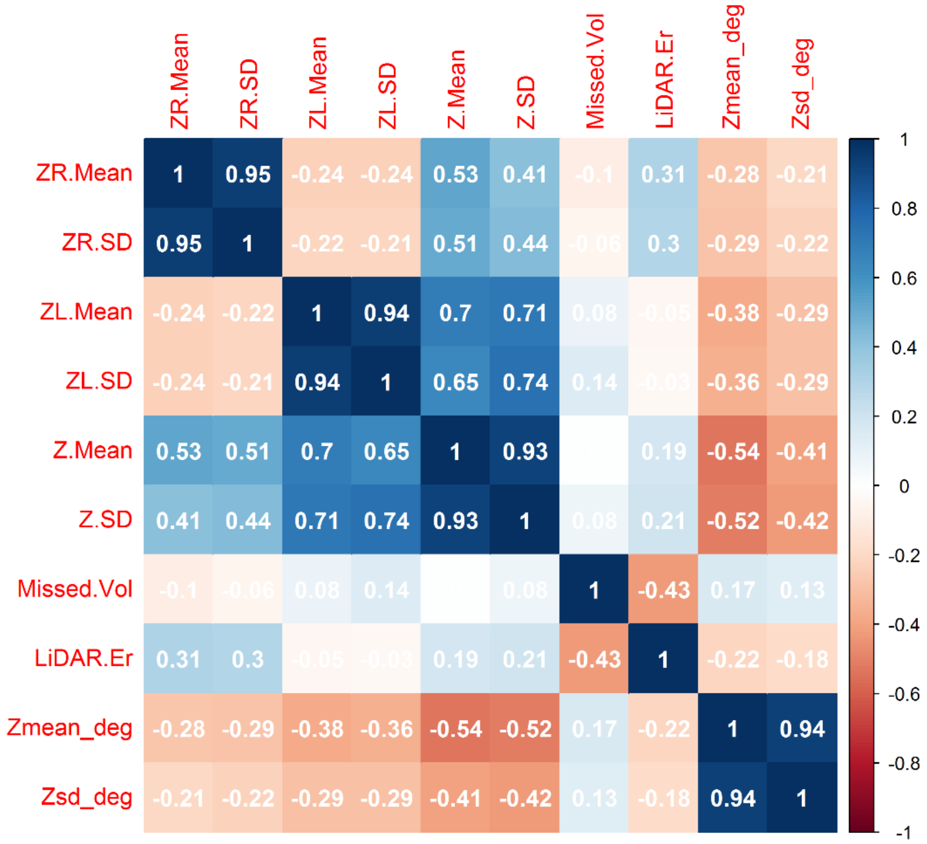
3. Correlation plot for the bank's slope to level of 10-years flood scenario.



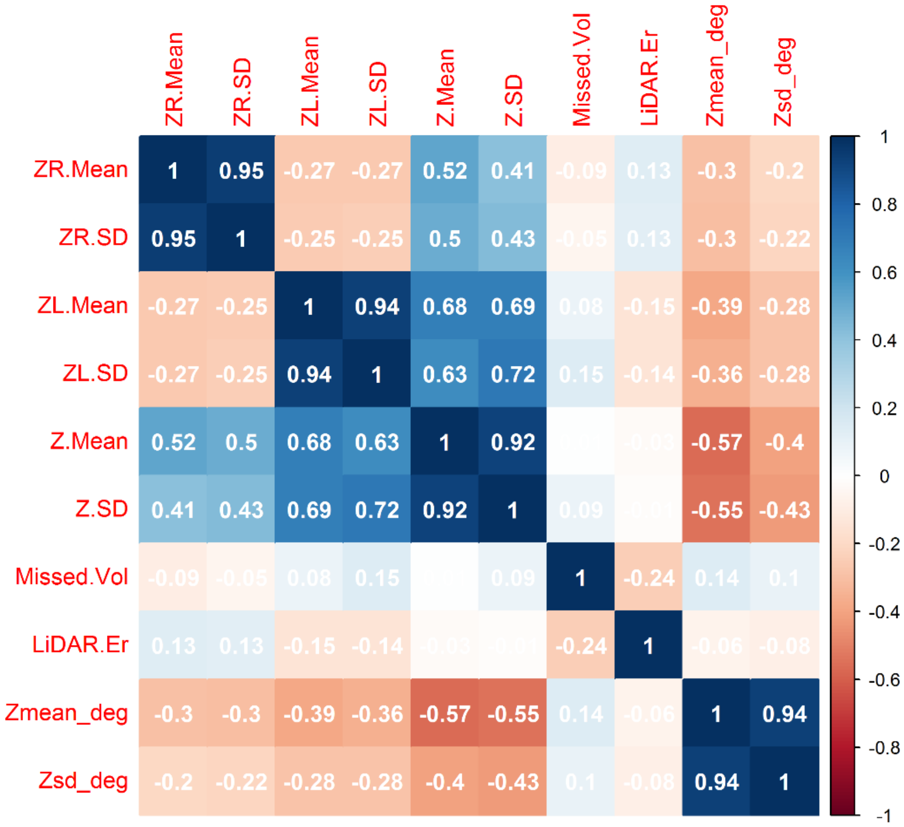
4. Correlation plot for the bank's slope to level of 10-years flood scenario.



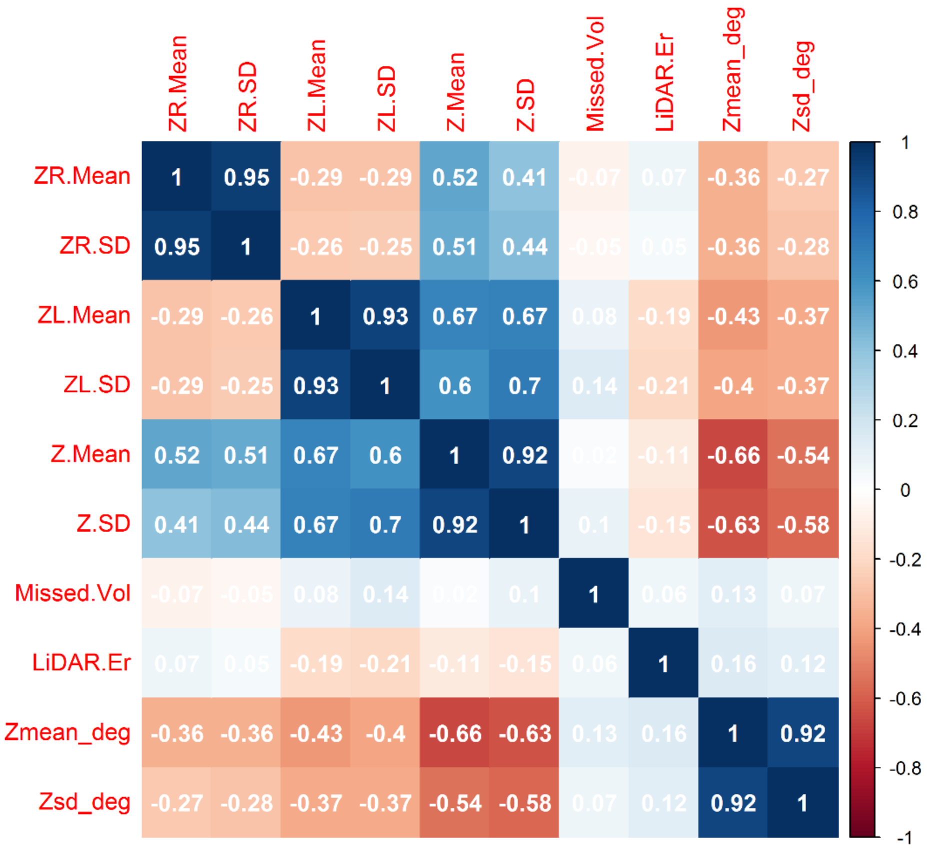
5. Correlation plot for the bank's slope to level of 10-years flood scenario.



6. Correlation plot for the bank's slope to level of 10-years flood scenario.

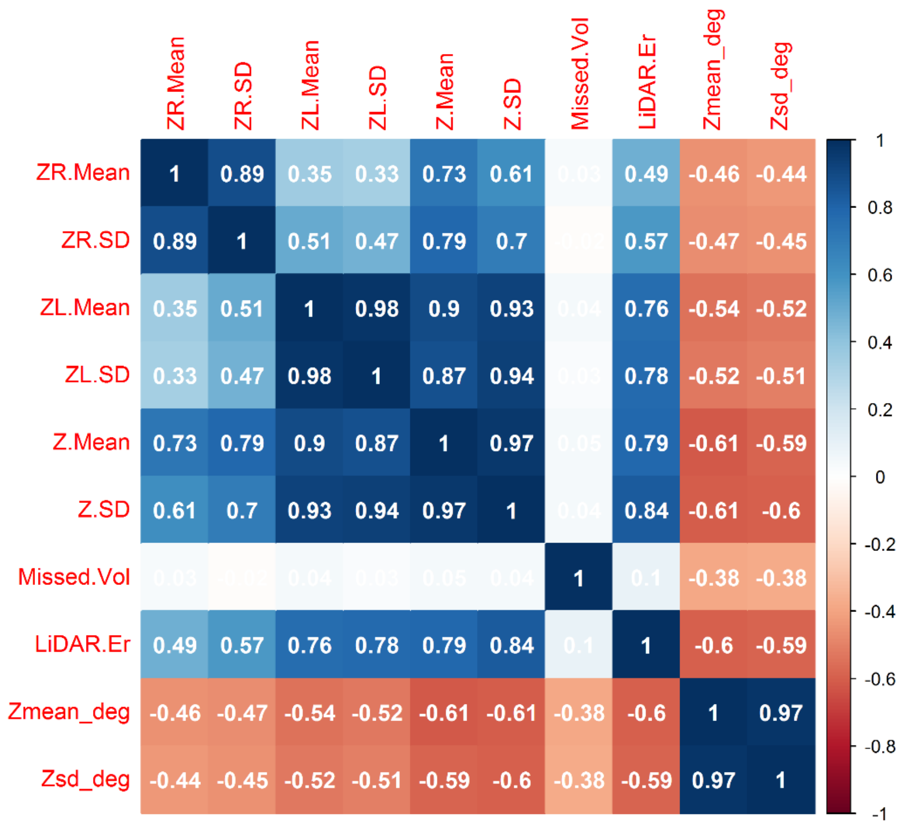


7. Correlation plot for the bank's slope to level of 10-years flood scenario.

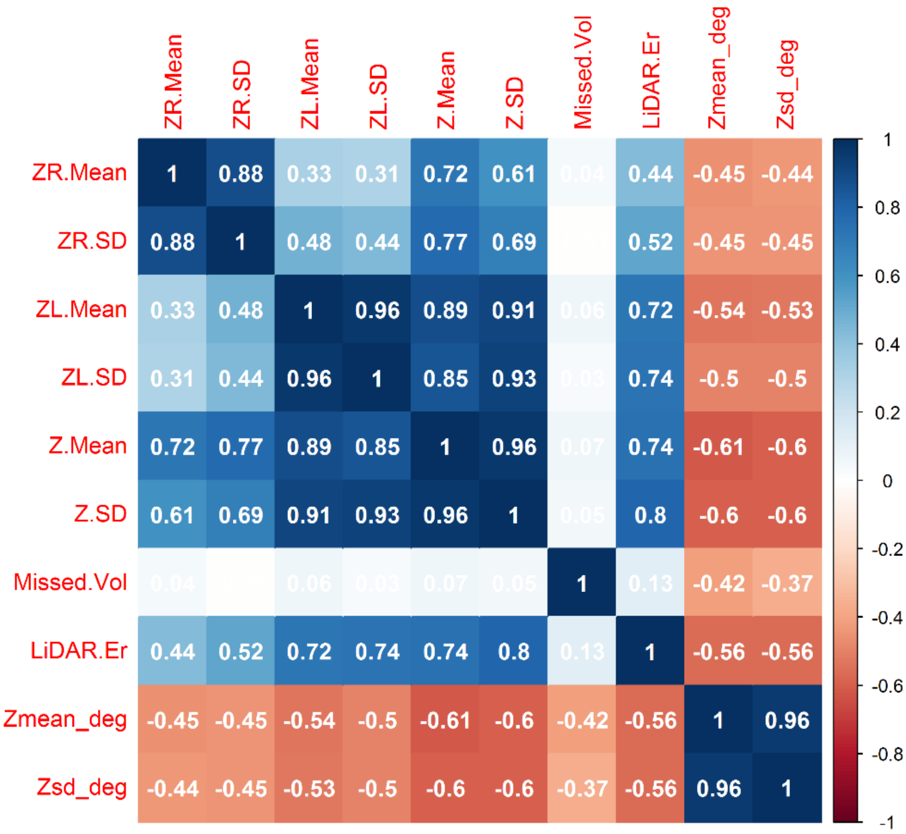


c. Upper Lærdal site

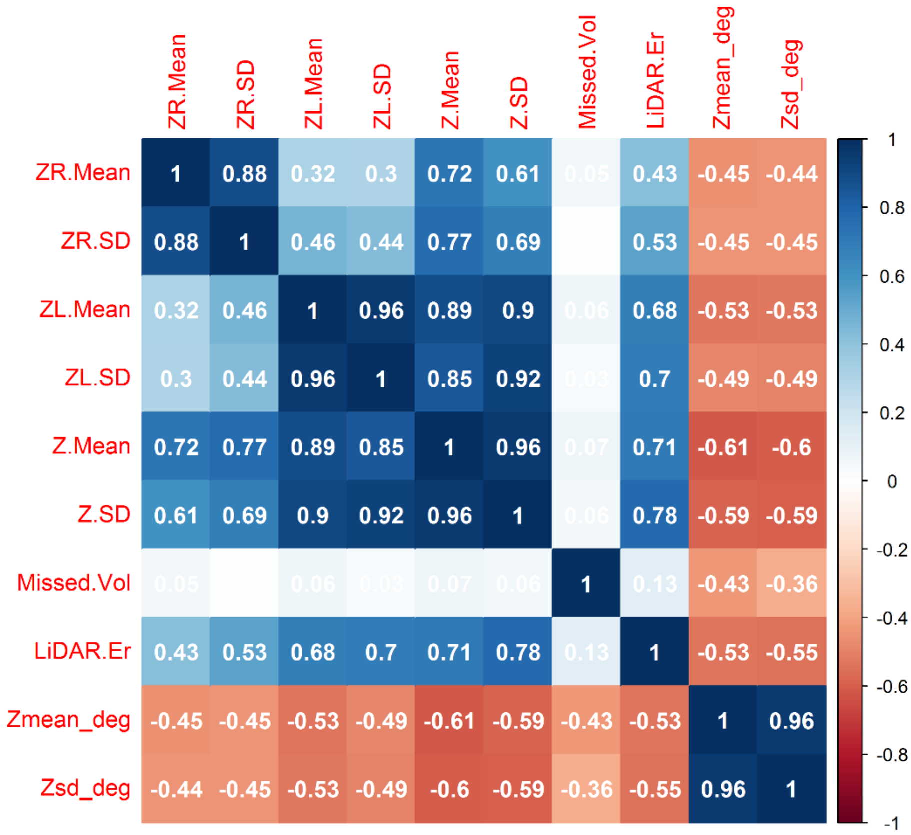
1. Correlation plot for the bank's slope to level of Mean flood scenario



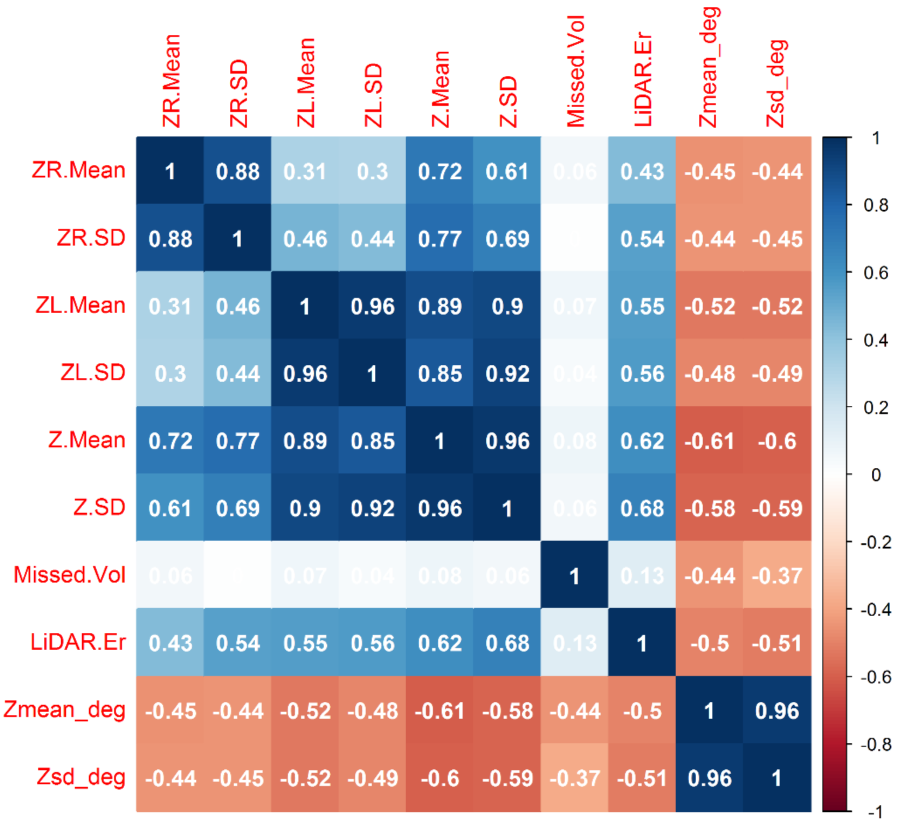
2. Correlation plot for the bank's slope to level of 10-years flood scenario



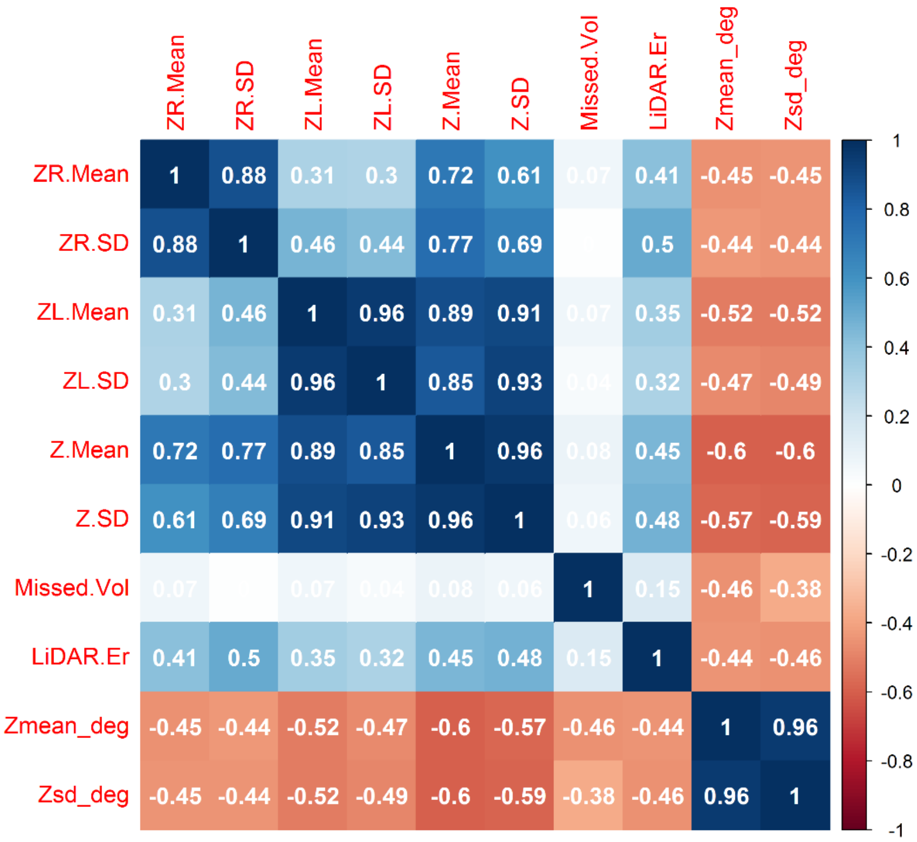
3. Correlation plot for the bank's slope to level of 20-years flood scenario



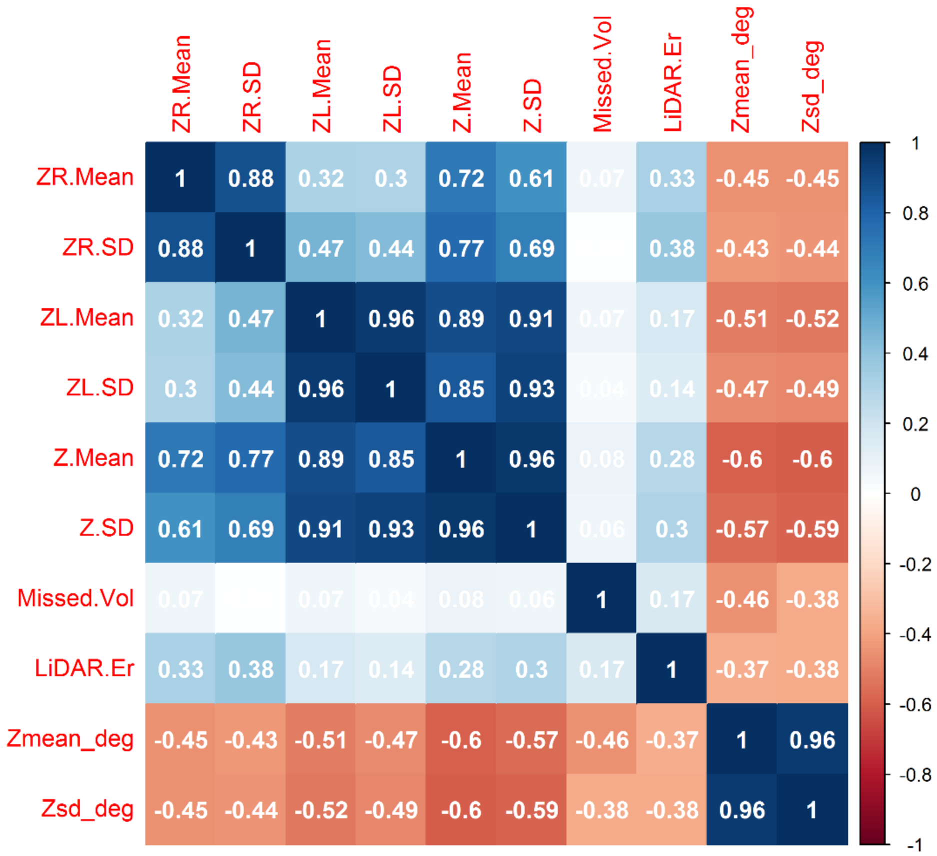
4. Correlation plot for the bank's slope to level of 50-years flood scenario



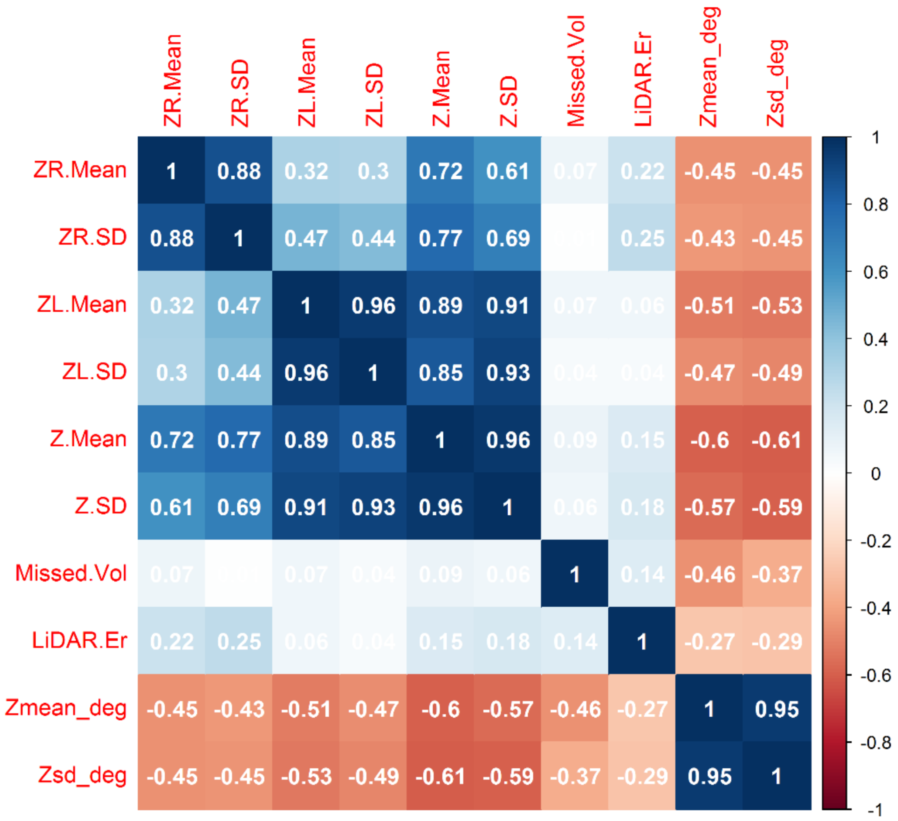
5. Correlation plot for the bank's slope to level of 100-years flood scenario



6. Correlation plot for the bank's slope to level of 200-years flood scenario

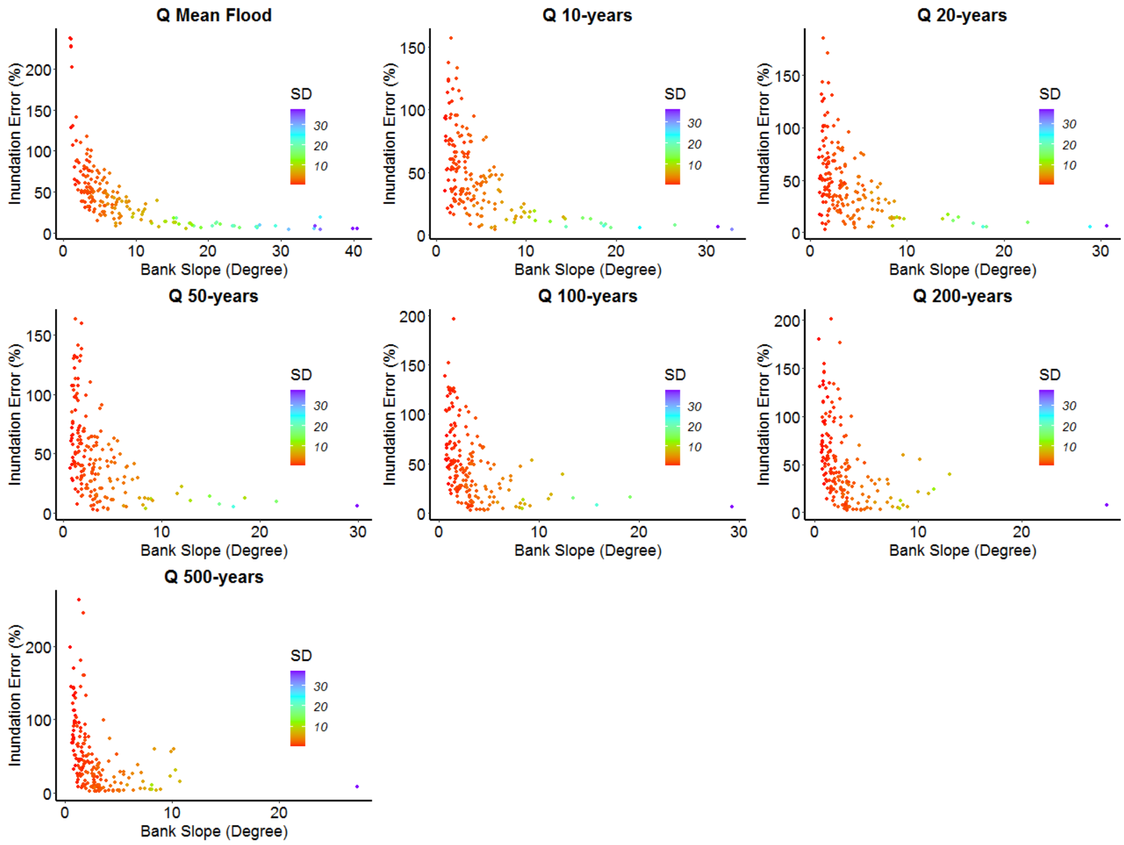


7. Correlation plot for the bank's slope to level of Mean flood scenario

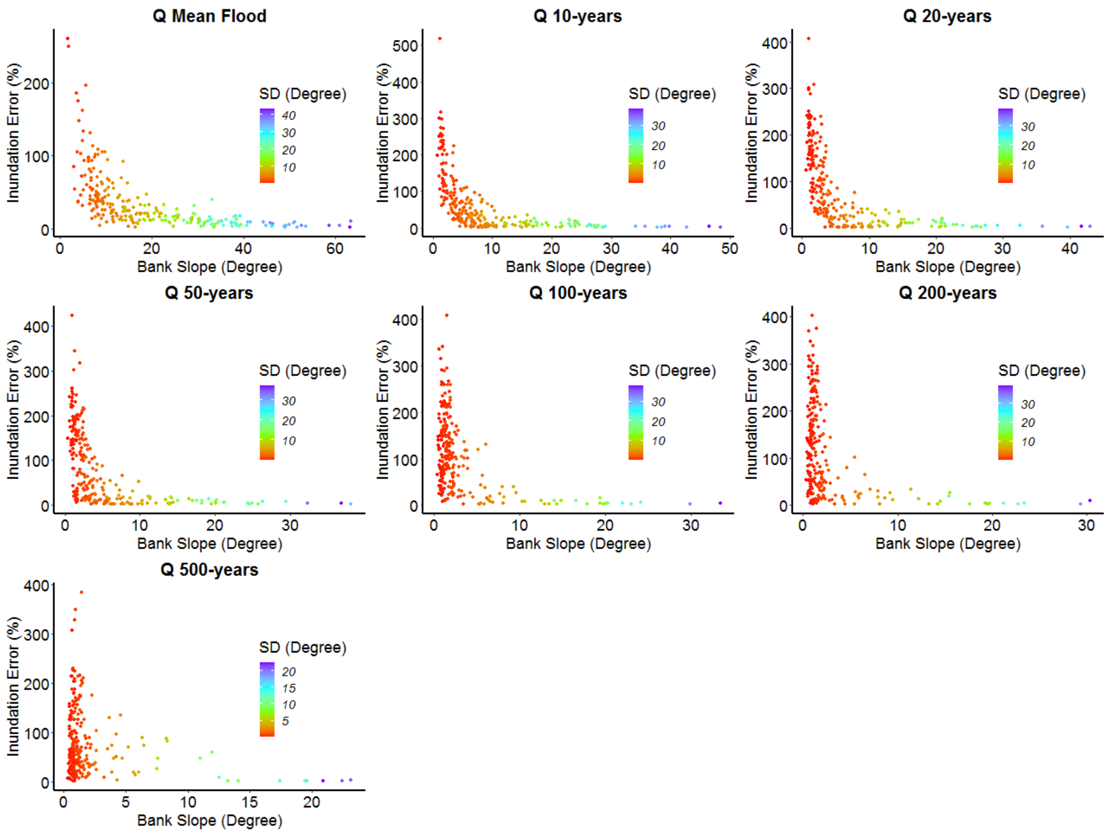


Bank's slope vs LiDAR Error for the rest of the sites

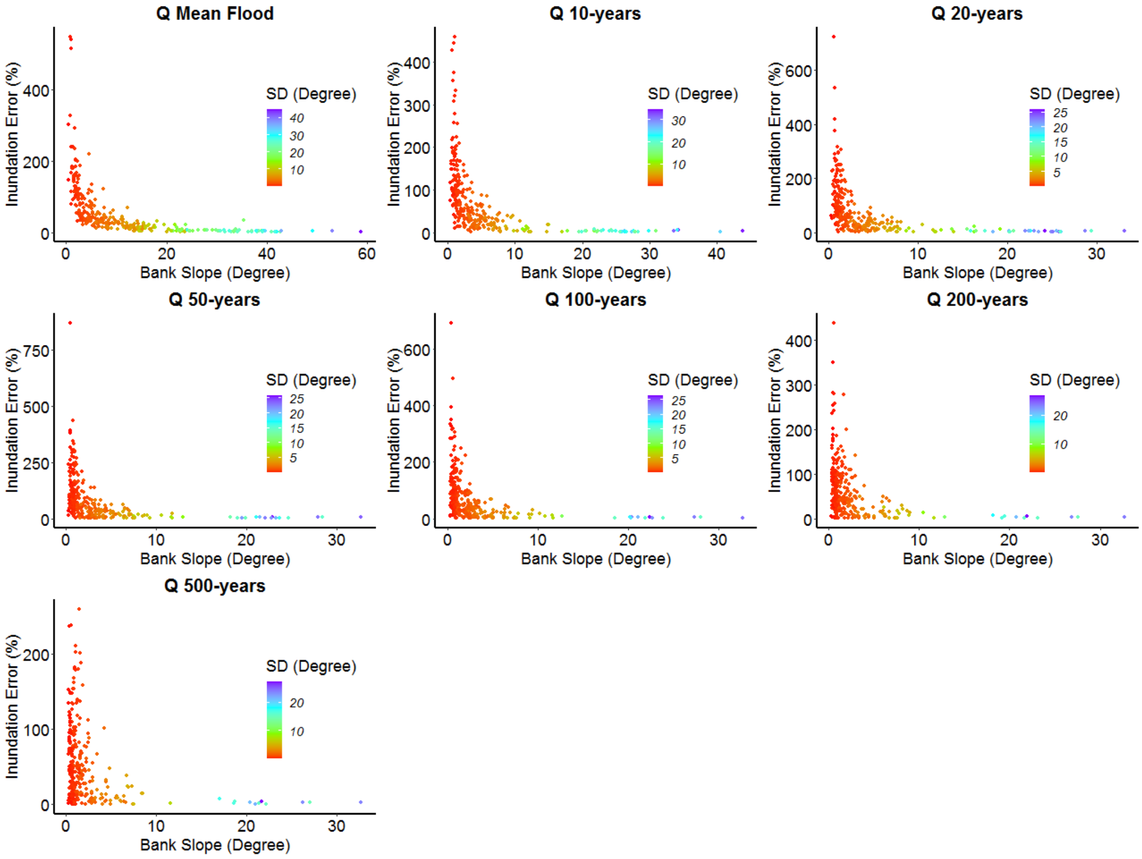
**Driva**



## Lower Lærdal

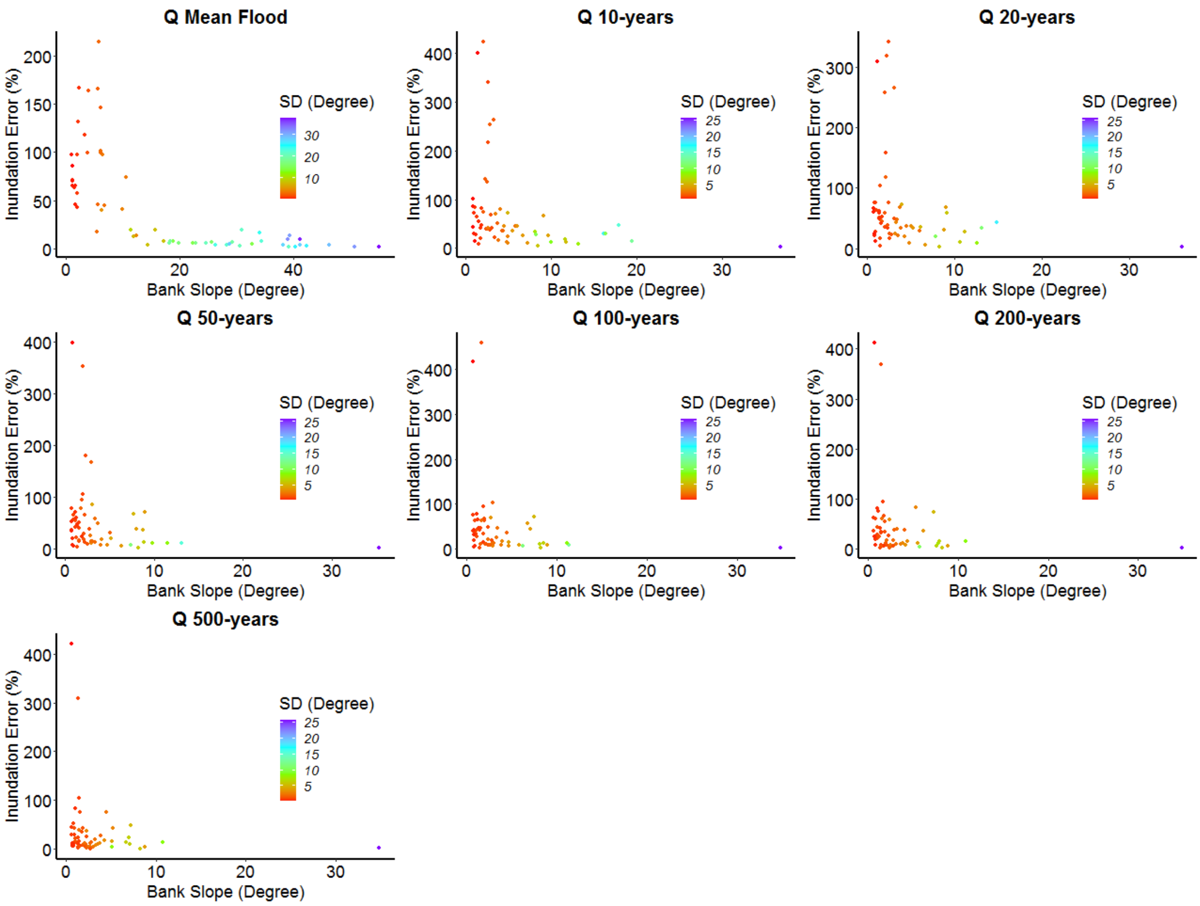


## Lower Surna

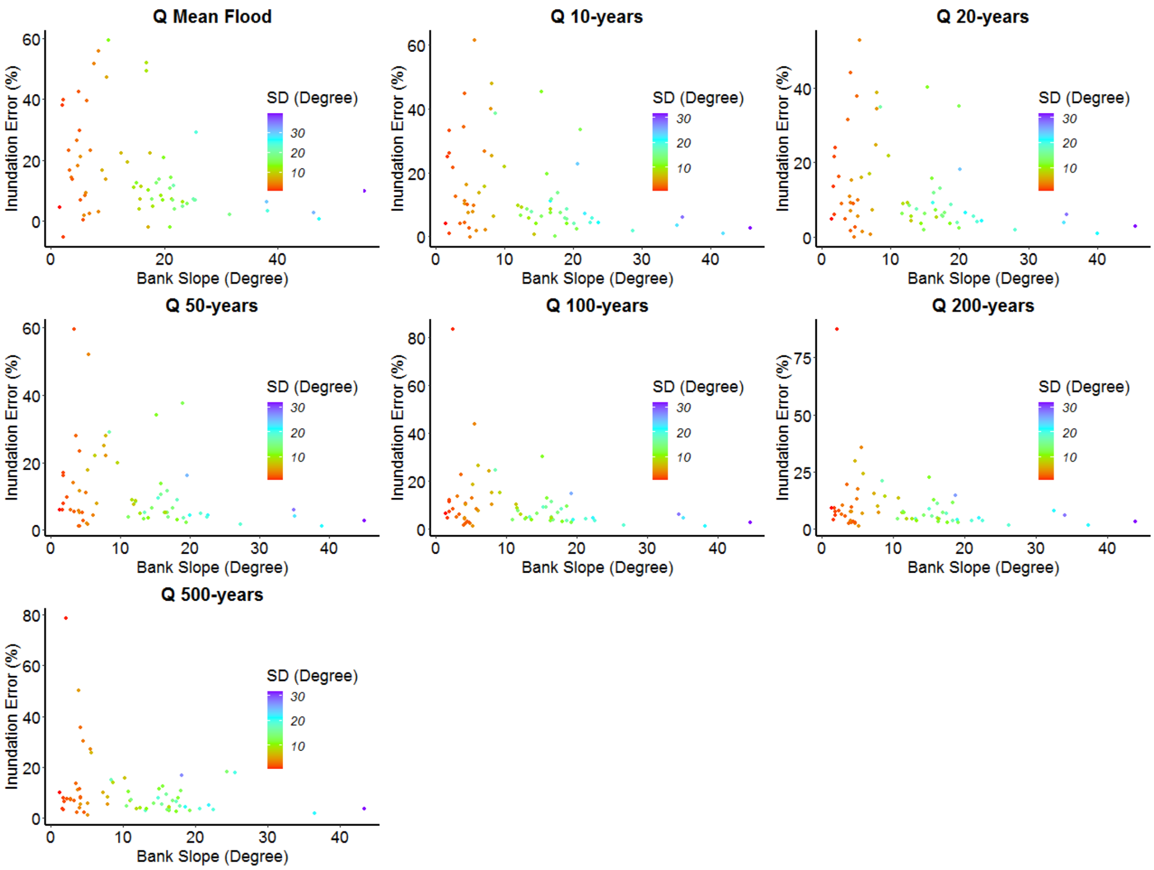


Q Mean Flood
Q 10-years
Q 20-years
Q 50-years
Q 100-years
Q 200-years
Q 500-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)

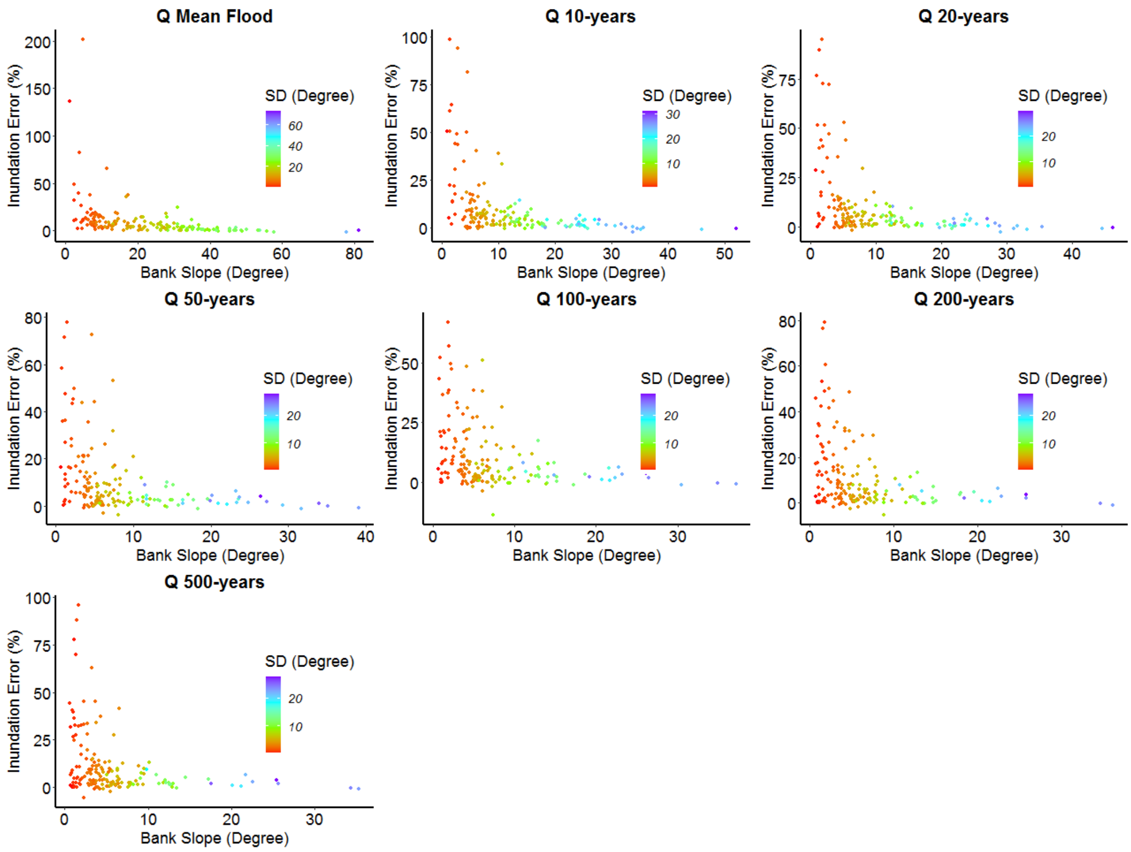
## Storåne


Q Mean Flood
Q 10-years
Q 20-years
Q 50-years
Q 100-years
Q 200-years
Q 500-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)

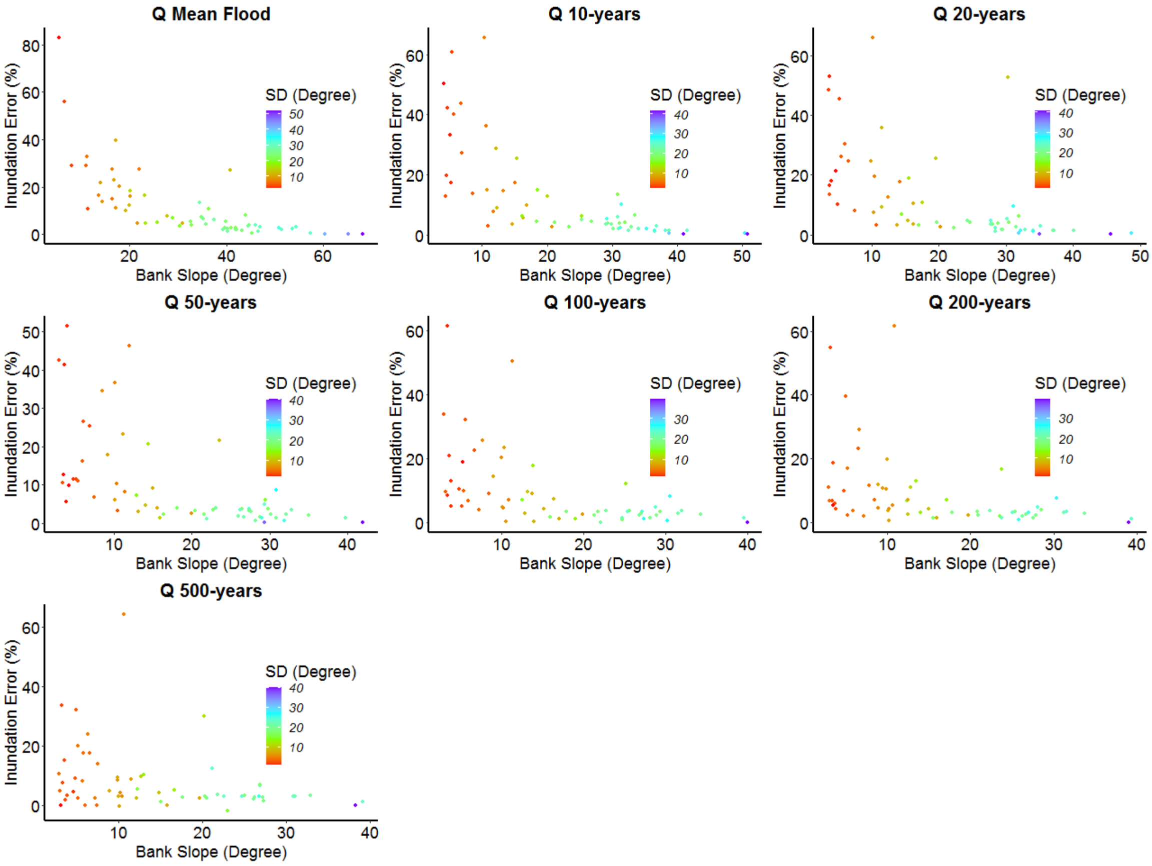
## Tokke


Q Mean Flood
Q 10-years
Q 20-years
Q 50-years
Q 100-years
Q 200-years
Q 500-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
30
20
10

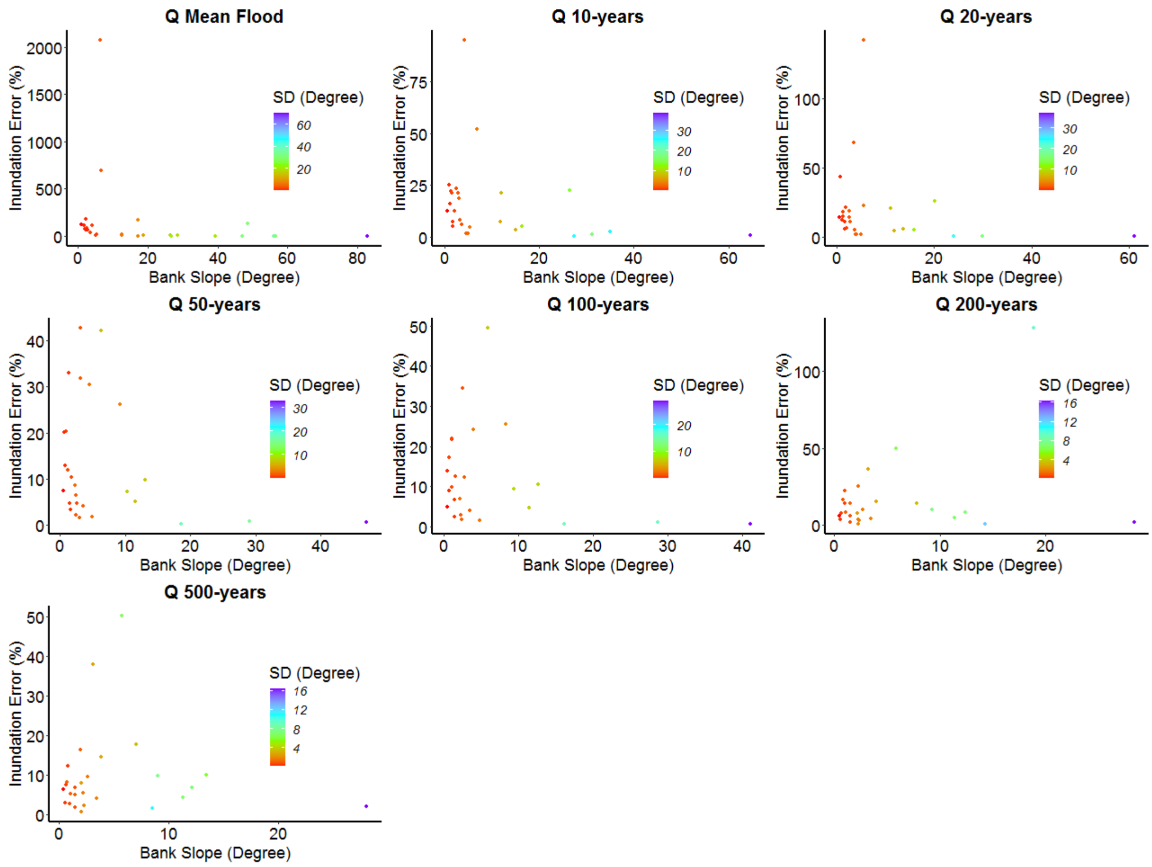
## Upper Surna


Q Mean Flood
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
Q 10-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
Q 20-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
Q 50-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
Q 100-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
Q 200-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)
Q 500-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)

## Sokna


Q Mean Flood
Q 10-years
Q 20-years
Q 50-years
Q 100-years
Q 200-years
Q 500-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)

## Gaua



Q Mean Flood
Q 10-years
Q 20-years
Q 50-years
Q 100-years
Q 200-years
Q 500-years
Inundation Error (%)
Bank Slope (Degree)
SD (Degree)

# Appendix C: LiDAR data information

| Site | Red LiDAR Project name | Green LiDAR Project name | Reach Extent | Date of Flight |
|---|---|---|---|---|
| *Driva* | Sunndal 2011-2013 | NVE Driva 2016 | River mouth to Grøa | 7/30/2013 |
| *Eidselva* | NDH Eid 5pkt 2017 | NVE Eidselva 5 pkt 2017 | DS Kviafossen to fjord | 8/18/2017 |
| *Gaula* | NDH Gauldal 2015 | NVE Gaula 2016 | Gaulfossen to fjord | 7/4/2015 |
| *Lower Lærdal* | Sogndal_Aurland_Lerdal 2014 | NVE Lærdal 2018 | Bridge DS Nivla to fjord | 4/26/2014 |
| *Lower Surna* | NDH Surnadal-Rindal 2pkt 2016 | Surna Statkraft project 2017 | DS Trollheimen PP to Skei | 10/16/2016 |
| *Storåne* | NDH Hol-l 5pkt 2018 | AHM_Ljungan, Hallingdal, and Tokkeåi 2015/2016 | From th Second bridge upstream the PP to Hovsfjorden | 6/6/2018 |
| *Tokke* | NDH Tokke 5pkt 2017 | AHM_Ljungan, Hallingdal, and Tokkeåi 2015/2016 | DS the confluence of Tokke and Dalåi to Bandak | 10/7/2017 |
| *Upper Lærdal* | NDH Lærdal 5pkt 2017 | NVE Lærdal 2018 | Sjurhaugfossen to the PP | 7/20/2017 |
| | Sogndal_Aurland_Lerdal 2014 | | | 4/26/2014 |
| *Upper Surna* | NDH Surnadal-Rindal 2pkt 2016 | Surna Statkraft project 2017 | from DS Rinda to Trollheimen PP | 10/16/2016 |
| *Sokna* | Midtre Gauldal 2014 | NVE Gaula 2016 | DS the confluence of Sokna and Hauka to Gaula | 6/10/2014 |
| *Gaua* | NDH Gauldal 2015 | NVE Gaula 2016 | The part included in the Green LiDAR | 7/4/2015 |

Appendix D: Correlation plot for the terrain analysis parameters