Evaluating Machine Learning and Statistical Models for Greenland Subglacial Bed Topography

ICMLA 2023

Katherine Yi[1], Angelina Dewar[2], Tartela Tabassum[3] , Jason Lu[4], Ray Chen[5],

Homayra Alam[3], Omar Faruque[3], Sikan Li[6], Mathieu Morlighem[7], **Jianwu Wang**[3]

[1] Department of Computer Science & Department of Statistics, Purdue University
[2] Department of Physics, University of Oregon
[3] Department of Information Systems, University of Maryland, Baltimore County
[4] College of Information Studies, University of Maryland, College Park
[5] Marriotts Ridge High School, Maryland
[6] Texas Advanced Computing Center, University of Texas at Austin
[7] Department of Earth Sciences, Dartmouth College

UMBC

iHARP NSF HDR INSTITUTE

Roadmap

1. Background
2. Methodology
3. Results
4. Conclusions
5. Acknowledgements
6. Q&A

https://th.bing.com/th/id/OIP.WRboS2aSvFY3w16TMnX8_gHaE8?w=278&h=185&c=7&r=0&o=5&dpr=1.6&pid=1.7; other image Australia Business Insider

What is the objective of this project?

Predict ice bed elevation at locations it has not been measured using features of ice surface derived via satellite for Greenland glaciers.

Why does this matter?

Improve our understanding ice beds which are changing in response to **climate change** and **reduce the uncertainty** in sea level rise projections which affect the world.

Background

Motivation

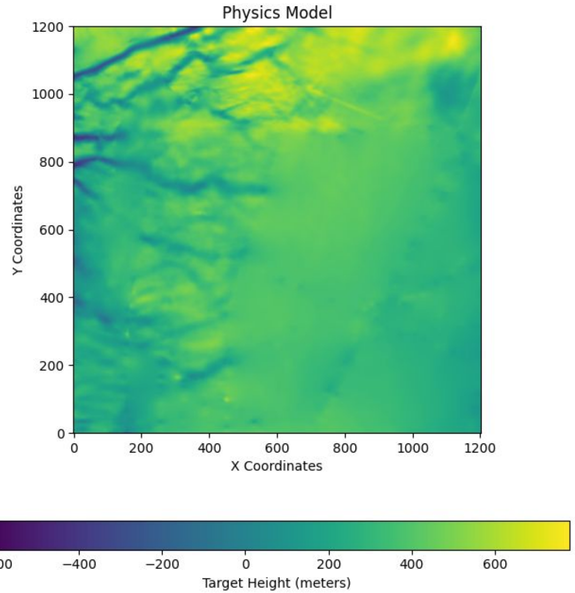
1. The bed controls the stability and vulnerability of the ice sheets.
2. Radar measurements remain sparse.
3. Consistently ranked #1 need by ice sheet modelers.

Literature Review

1. Leong and Horgan (2020) explored the application of machine learning, particularly CNNs, to **predict bedrock topography in Antarctica**.
2. Morlighem explored **physics based bed machines** for Greenland (2014 & 2017).
3. Liu-Schiaffini and colleagues (2022) explored the general use of **deep learning on radar grams of ice sheets.**

BedMachine Map, generated using physics

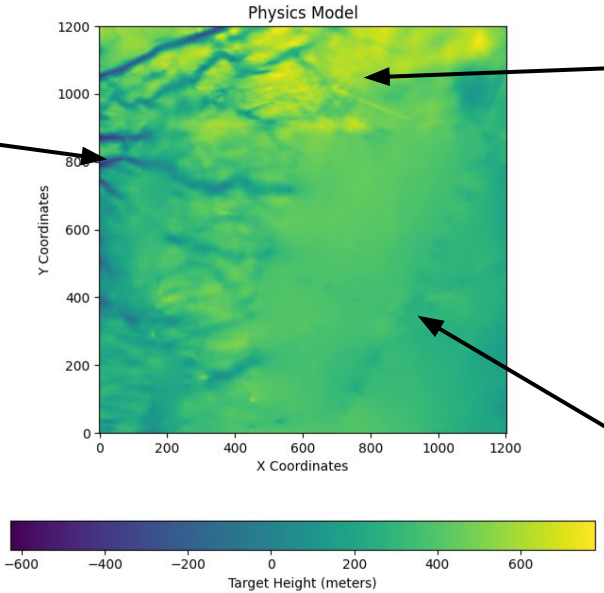
(Morlighem, 2017)



BedMachine Map, generated using physics

(Morlighem, 2017)

Dark Blue is low terrain area cut out by flowing water.
Goal: Emphasize these veins more.

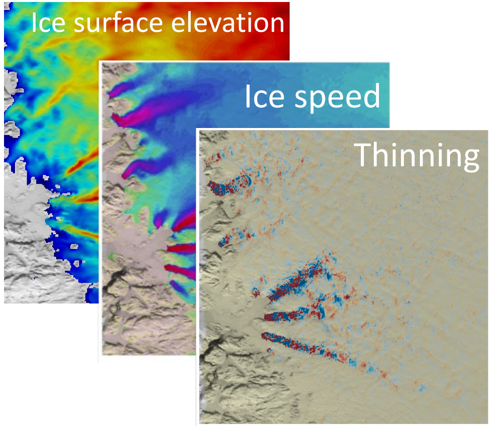


Yellow is high mountain area.

Fuzzy area is unclear predictions.
Goal: Make clearer predictions.

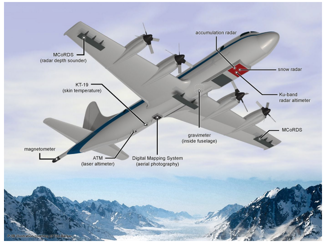
Data Background (632,706 labeled data points)

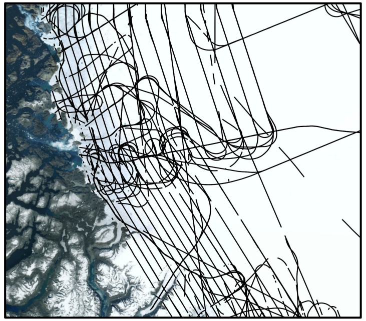
1200 x 1200 square grids with grid size of 150×150 m^2



Surface variables

1. Surface Elevation (m)
2. Ice thinning rates (m/yr)
3. Surface mass balance (m)
4. Ice flow velocity horizontal (m/yr)
5. Ice flow velocity vertical (m/yr)





Ice Bed Elevation with limited observations

UMBC

iHARP NSF HDR INSTITUTE

Challenges

1. Our **data is sparse** because data collection is expensive and coordinates across our **datasets' coordinates do not align**.

2. **Identify a model** that can effectively capture sequential and spatial data.

Challenges

1. Our **data is sparse** because data collection is expensive and coordinates across our **datasets' coordinates do not align**.

2. **Identify a model** that can effectively capture sequential and spatial data.

Solutions

1. Data **interpolation** to match data sources from satellite analysis and radar together.

2. Understand and **reframe the project goal** to fit the environmental context.

Methodology

Overview → Interpolation & Modeling

Methodology Overview

Interpolation & Modeling

Applied

- 3 interpolation approaches
- 10 Models
  - 4 machine learning
  - 3 statistical
  - 3 hybrid

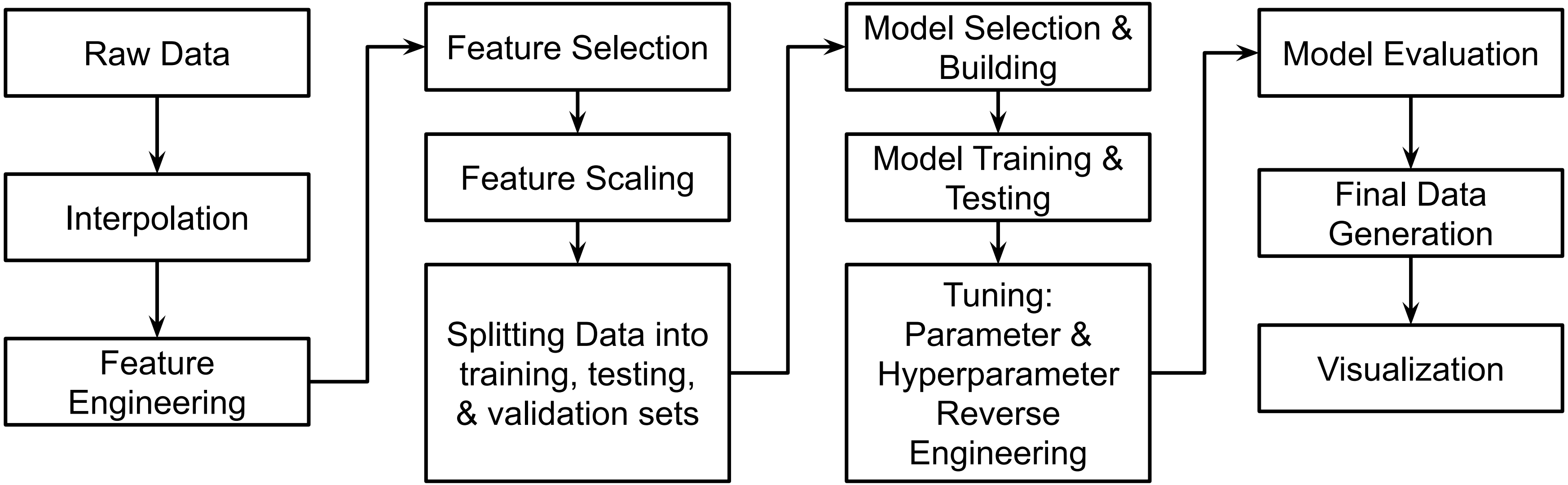
Ablation analysis

- Feature Engineering
- Preprocessing for modeling (ex: train/test splits)
- Interpolation Methods
- Best Models

Pipeline

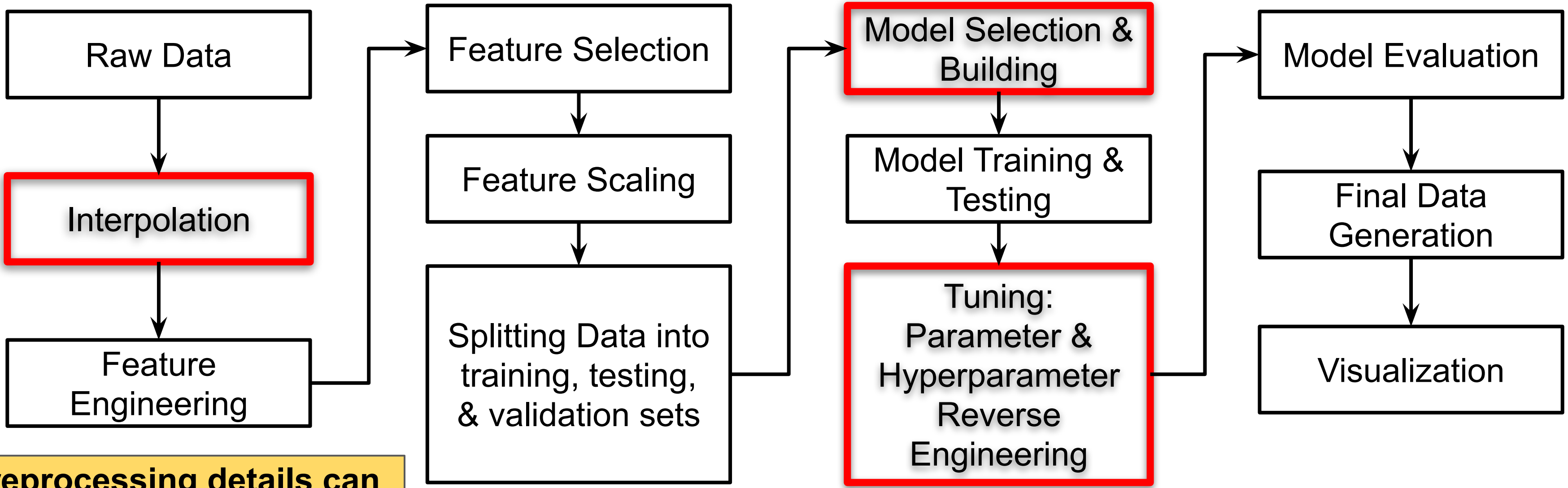
Methodology Overview

Interpolation & Modeling



Methodology Overview

Interpolation & Modeling

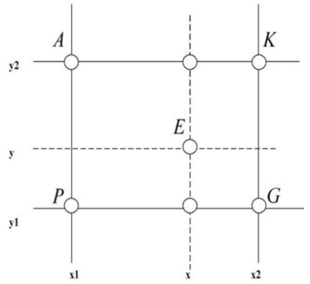


Preprocessing details can be found in our paper

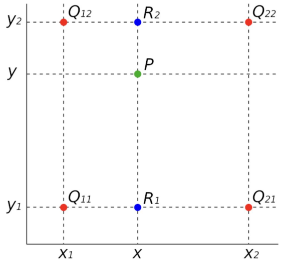
Interpolation

Approaches explored to put our data together

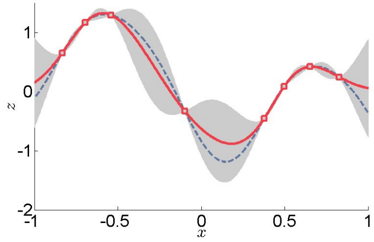
- Nearest Neighbor: gets the values of nearest surface observation
- Bilinear interpolation: takes the weighted average of values of surface feature at 4 nearest neighbors
- Universal Kriging: weighted sum of values of surface feature at all locations within a defined neighborhood



Nearest Neighbors Interpolation example



Bilinear Interpolation example



1D Universal Kriging Interpolation example

https://x-engineer.org/bilinear-interpolation/; Ref:Rukundo, Olivier and Hanqiang Cao. "Nearest Neighbor Value Interpolation." ArXiv abs/1211.1768 (2012): n. pag.; By Antro5 at English Wikipedia, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=52740780

Modeling

Modeling success comes from understanding your objective in context.

What is our context?

What do we know about this context? (Generally)

How does the translate to model requirements?

Modeling

Modeling success comes from understanding your objective in context.

What is our context?

Terrain in Greenland

What do we know about this context? (Generally)

How does the translate to model requirements?

Modeling

Modeling success comes from understanding your objective in context.

What is our context?

Terrain in Greenland

What do we know about this context? (Generally)

1. **No guarantee** of consistency
2. No guarantee of normality
3. Terrain has localized trends (ex: hills)
4. Different regions have relationships (ex: beach into the ocean)
5. Terrain changes over time given external effects (ex: dirt erodes with rain)

How does the translate to model requirements?

Modeling

Modeling success comes from understanding your objective in context.

What is our context?

Terrain in Greenland

What do we know about this context? (Generally)

1. **No guarantee** of consistency
2. No guarantee of normality
3. Terrain has localized trends (ex: hills)
4. Different regions have relationships (ex: beach into the ocean)
5. Terrain changes over time given external effects (ex: dirt erodes with rain)

How does the translate to model requirements?

Model must capture:

1. **Local trends**
2. **Larger spatial trends and relationships**

Model must have **flexibility.**

What models were explored?

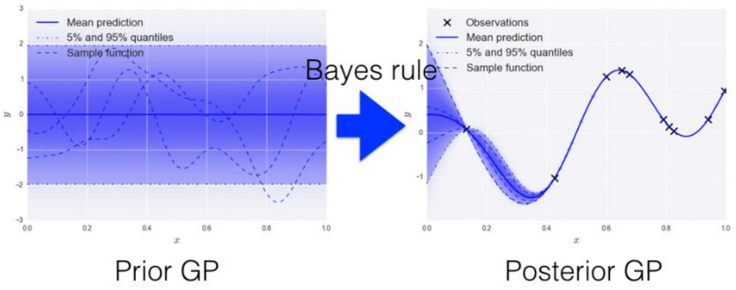
Models Explored

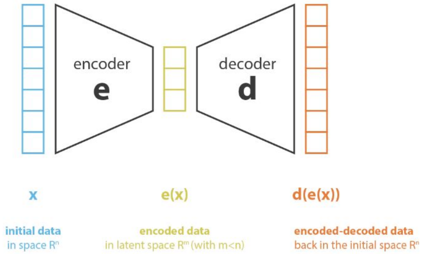
Machine learning models

- **Extreme Gradient Boosting (XGBoost)**
- Dense Neural Network
- Long-Short Term Memory (LSTM)
- Dense + LSTM
- Kriging Residual Learning
- Variational Autoencoder (VAE)
- VAE + XGBoost

Probabilistic models ($$$$)

- Gaussian Process Regression (GPR)
- Spatio-Temporal Gaussian Processing (STGP)
- Kriging First Pass Prediction



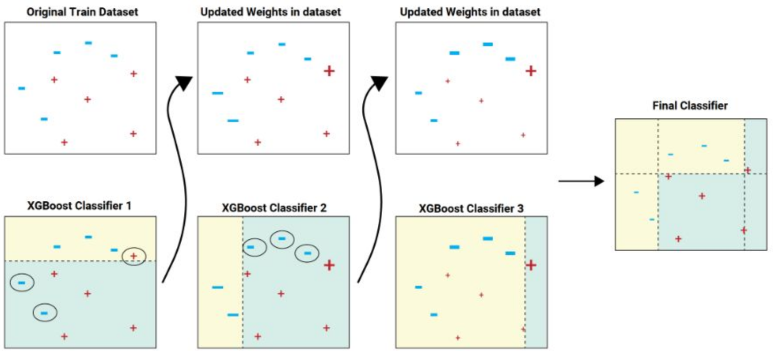


Gaussian Processes regression: basic introductory example — scikit-learn 1.3.2 documentation

XGBoost

Parameters Tuned:

- Depth of decision tree
- Number of boosting rounds
- Minimum child weight
- Subsample for training
- Learning rate



Reverse Engineering Parameters

Model Selection

Set Parameters, Training, & Testing

Generate Data

Visualizations

Testing

Understand how parameters affect the map

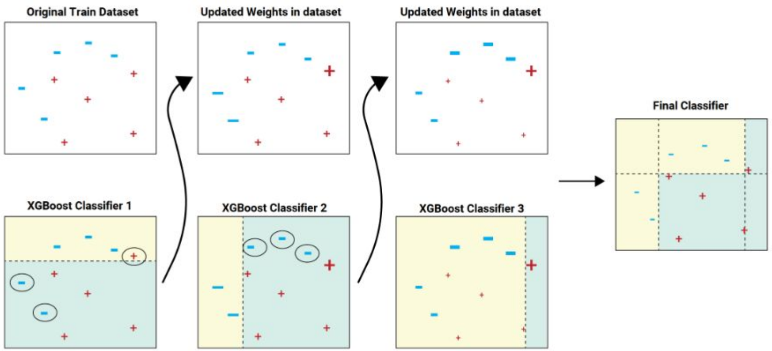
Tune the parameters to optimize for our objective of a clear map - not to minimize loss metrics

Repeat Tuning and Optimize Based on Output

XGBoost

Parameters Tuned:

- Depth of decision tree
- Number of boosting rounds
- Minimum child weight
- Subsample for training
- Learning rate



Reverse Engineering Parameters

Model Selection

Set Parameters, Training, & Testing

Generate Data

Visualizations

Testing

Understand how parameters affect the map

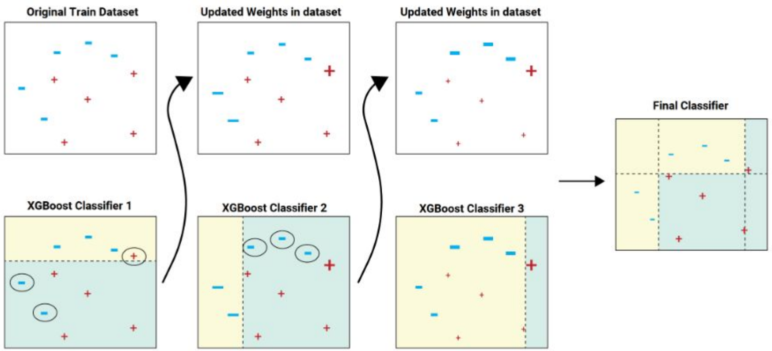
Tune the parameters to optimize for our objective of a clear map - not to minimize loss metrics

Repeat Tuning and Optimize Based on Output

XGBoost

Parameters Tuned:

- Depth of decision tree
- Number of boosting rounds
- Minimum child weight
- Subsample for training
- Learning rate



Reverse Engineering Parameters

Model Selection

Set Parameters, Training, & Testing

Generate Data

Visualizations

Testing

Understand how parameters affect the map

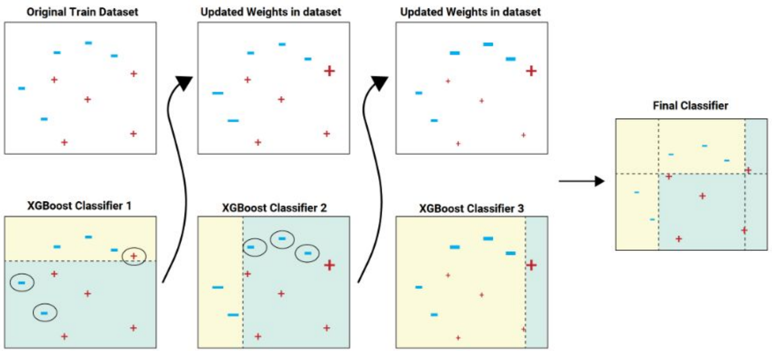
Tune the parameters to optimize for our objective of a clear map - not to minimize loss metrics

Repeat Tuning and Optimize Based on Output

XGBoost

Parameters Tuned:

- Depth of decision tree
- Number of boosting rounds
- Minimum child weight
- Subsample for training
- Learning rate



Reverse Engineering Parameters

Model Selection

Set Parameters, Training, & Testing

Generate Data

Visualizations

Testing

Understand how parameters affect the map

Tune the parameters to optimize for our objective of a clear map - not to minimize loss metrics

Repeat Tuning and Optimize Based on Output

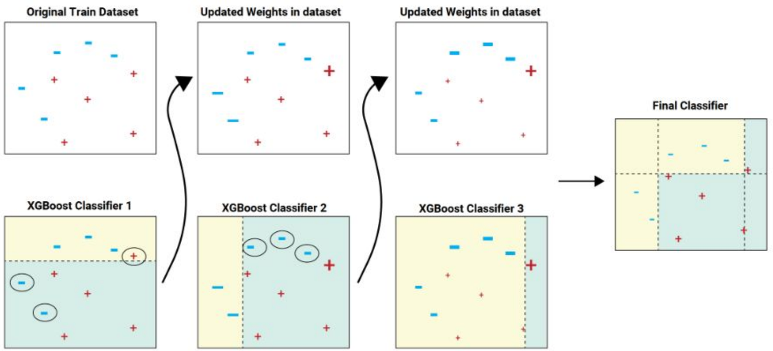
XGBoost

Parameters Tuned:

- Depth of decision tree
- Number of boosting rounds
- Minimum child weight
- Subsample for training
- Learning rate

Map Characteristics Affected:

- **Smoothness of the map across localized areas**
- **Clarity in contrasting high and low terrain**



Reverse Engineering Parameters

Model Selection

Set Parameters, Training, & Testing

Generate Data

Visualizations

Testing

Understand how parameters affect the map

Tune the parameters to optimize for our objective of a clear map - not to minimize loss metrics

Repeat Tuning and Optimize Based on Output

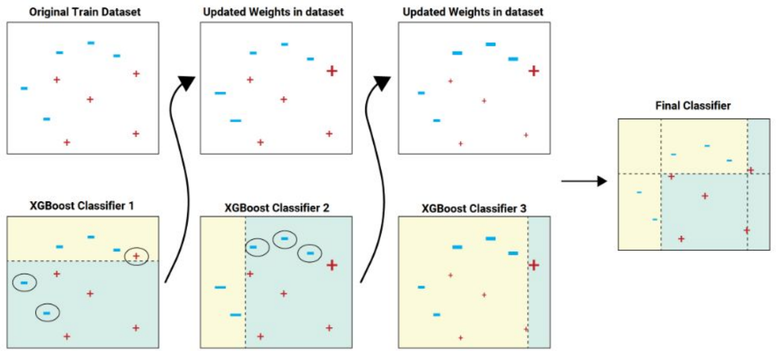
XGBoost

Parameters Tuned:

- Depth of decision tree
- Number of boosting rounds
- Minimum child weight
- Subsample for training
- Learning rate

Map Characteristics Affected:

- Smoothness of the map across localized areas
- Clarity in contrasting high and low terrain



Reverse Engineering Parameters

Model Selection → Set Parameters, Training, & Testing → Generate Data → Visualizations → Testing → Understand how parameters affect the map → Tune the parameters to optimize for our objective of a clear map - not only to minimize error metrics

Repeat Tuning and Optimize Based on Output

Metrics & Results

https://th.bing.com/th/id/OIP.iLn5SWkiIZMC5kvQMHrEvAHaE8?w=251&h=180&c=7&r=0&o=5&dpr=1.6&pid=1.7

How to measure success?

1. Root Mean Squared Error (RMSE)
   Measures difference in predicted and known data considering the size of the error and outliers

2. Mean Absolute Error (MAE)
   Measures difference in predicted and known data, but is not as sensitive to outliers

3. Coefficient of Determination (R^2)
   Assessing goodness of fit and ranges from 0-1.
   $1 \longrightarrow$ extremely correlated; $0 \longrightarrow$ poor relationship

4. Terrain Roughness Index (TRI)
   Ranges from 0-280+ meters world wide.
   Lower score is flatter terrain and higher score is more rigorous terrain.
   Nunn and Puga (2012) report Greenland as 41m TRI measured in 1996.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (predicted_i - actual_i)^2}{N}}$$

$$MAE = \frac{\sum_{i=1}^{N} |actual_i - predicted_i|}{N}$$

$$R^2 = \frac{SSR}{SST} = \frac{\sum(predicted_i - actual)^2}{\sum(actual_i - \bar{y})^2}$$

TRI calculation for a single cell:

1. Calculate difference between the center cell and surrounding 8 cells.
2. Average the squared differences.
3. Finally, take the square root.

https://ourworldindata.org/grapher/terrain-ruggedness-index

Results of Different Models

Only reporting validation data metrics with nearest neighbor based data interpolation

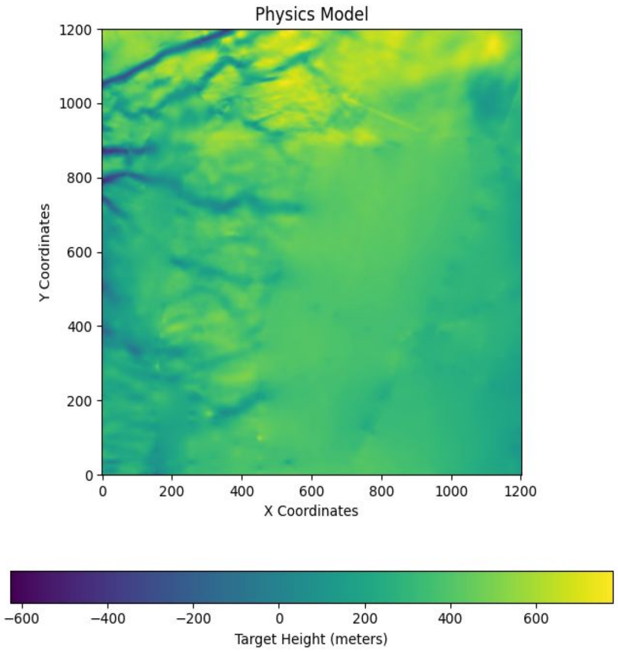
Model	Type	RMSE	MAE	R^2
XGBoost	Machine learning	**32.680**	22.273	**0.967**
BedMachine	Physics	71.554	50.422	0.842
LSTM + MLP	Hybrid	89.937	60.328	0.783
LSTM	Deep learning	101.630	74.522	0.682
MLP (Dense)	Deep learning	104.475	74.381	0.663
VAE	Deep learning	106.884	83.798	0.648
VAE + XGB	Hybrid	129.760	100.035	0.481
Kriging Residual	Hybrid	136.650	8.122	0.424
GPR	Probabilistic	150.086	97.855	0.293
Kriging	Probabilistic	188.948	**7.800**	-0.103
STGP	Probabilistic	225.772	126.819	-0.580

Ablation Study: Results for Different Interpolation Methods

Only reporting validation data metrics with XGBoost model

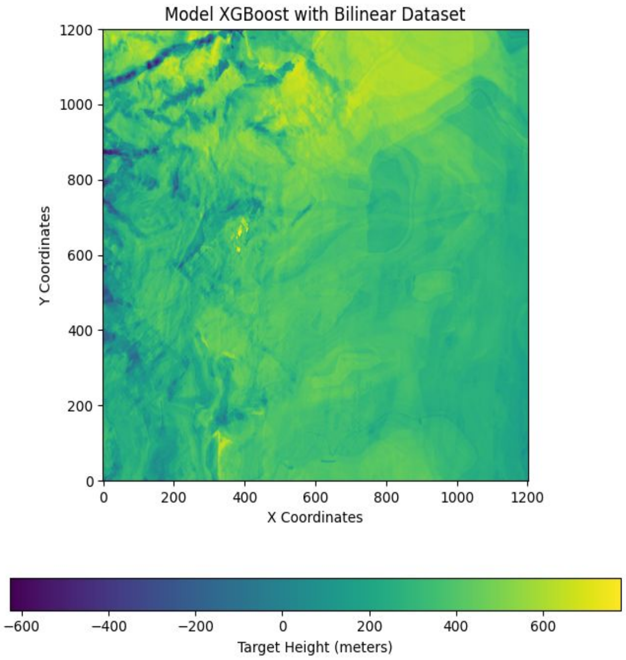
Interpolation	RMSE	MAE	R^2	TRI Score	Memory (GB)
Kriging	**27.099**	**17.947**	**0.977**	15.926	~320 in total with 5 nodes on one HPC environment
Nearest neighbor	32.680	22.273	0.967	14.178	**12** with 1 instance on google collab
Bilinear	28.085	18.549	0.976	14.682	**12** with 1 instance on google collab

Final Map, generated using Bilinear Interpolation & XGBoost



VS.

Most notable improvement is in the **clarity of the newly generated predictions**.



Conclusions

https://th.bing.com/th/id/OIP.hXij13CYWY48_OT4wiLp4gHaE8?w=217&h=180&c=7&r=0&o=5&dpr=1.6&pid=1.

Conclusion

The team tested many models with data interpolated in different ways. RMSE, MAE, R^2 & TRI metrics were tracked to assess performance of models.

Top performing models and interpolation:

1. XGBoosting with the Bilinear interpolation dataset
2. XGBoosting with the Kriging interpolated dataset

With the provided data, the XGBoosting model with Bilinear interpolation can be applied to Greenland topography and produce predictions of the topography with low error and high R2.

Success comes from the details in understanding our context.
Take the time to understand, plan, and execute from the beginning.

Acknowledgements

We would like to thank all of our collaborators for their guidance and domain expertise throughout the project.

We would also like to thank NSF, UMBC, HPCF, IHARP, and Big Data REU

For access to the GitHub and technical report, navigate to the Big Data REU Site

Big Data REU Website

What’s on your mind?

Q&A

Email: klyi@purdue.edu

LinkedIn

References

[1] M. Morlighem, C. N. Williams, E. Rignot, L. An, J. E. Arndt, J. L. Bamber, G. Catania, N. Chauche, J. A. Dowdeswell, B. Dorschel ´et al., “Bedmachine v3: Complete bed topography and ocean bathymetry mapping of greenland from multibeam echo sounding combined with mass conservation,” Geophysical research letters, vol. 44, no. 21, pp.11–051, 2017.
[2] W. J. Leong and H. J. Horgan, “Deepbedmap: A deep neural network for resolving the bed topography of antarctica,” The Cryosphere, vol. 14, no. 11, pp. 3687–3705, 2020.
[3] “Github repository for predicting ice-bed topography using predic-tive modeling.” https://github.com/big-data-lab-umbc/big-data-reu/tree/main/2023-projects/team-1, [Online; Accessed: 2023-07-30].
[4] S. J. Riley, S. D. DeGloria, and R. Elliot, “Index that quantifies topographic heterogeneity,” intermountain Journal of sciences, vol. 5, no. 1-4, pp. 23–27, 1999.
[5] M. Morlighem, E. Rignot, J. Mouginot, H. Seroussi, and E. Larour, “Deeply incised submarine glacial valleys beneath the greenland ice sheet,” Nature Geoscience, vol. 7, no. 6, pp. 418–422, 2014.
[6] M. B. Lythe and D. G. Vaughan, “BEDMAP: Anew ice thickness and subglacial topographic model of Antarctica.” J. Geophys. Res., vol. 106 (B6), pp. 11,335–11,351, 2001.
[7] P. e. a. Fretwell, “Bedmap2: improved ice bed, surface and thickness datasets for Antarctica,” Cryosphere, vol. 7, no. 1, pp. 375–393, 2013.
[8] P. Goovaerts, “Geostatistical software,” in Handbook of applied spatial analysis: Software tools, methods and applications. Springer, 2009, pp.125–134.
[9] M. Liu-Schiaffini, G. Ng, C. Grima, and D. Young, “Ice thickness from deep learning and conditional random fields: application to ice-penetrating radar data with radiometric validation,” IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1–14, 2022.
[10] C. Schoof, “Ice sheet grounding line dynamics: Steady states, stability, and hysteresis,” J. Geophys. Res., vol. 112, no. F03S28, pp. 1–19, JUL14 2007.
[11] I. Howat, A. Negrete, and B. Smith., “Measures greenland ice mapping project (gimp) digital elevation model from geoeye and worldview imagery, version 1,” 2017. [Online]. Available: https://nsidc.org/data/NSIDC-0715/versions/1
[12] J. Mouginot, E. Rignot, B. Scheuchl, and R. Millan, “Comprehensive annual ice sheet velocity mapping using landsat-8, sentinel-1, and radarsat-2 data,” Remote Sensing, vol. 9, no. 4, p. 364, 2017.
[13] B. Smith, S. Adusumilli, B. M. Csatho, D. Felikson, H. A. Fricker, A. Gardner, N. Holschuh, J. Lee, J. Nilsson, F. S. Paolo, M. R. Siegfried, T. Sutterley, and the ICESat-2 Science Team. (2021) Atlas/icesat-2 l3a land ice height, version 5. [Online]. Available:https://nsidc.org/data/ATL06/versions/5
[14] J. M. V. Wessem and M. K. Laffin. (2020, Feb) Regional atmospheric climate model (racmo2), version 2.3p2. [Online]. Available: https://doi.org/10.5281/zenodo.3677642
[15] J. A. MacGregor, L. N. Boisvert, B. Medley, A. A. Petty, J. P. Harbeck, R. E. Bell, J. B. Blair, E. Blanchard-Wrigglesworth, E. M. Buckley, M. S. Christoffersen et al., “The scientific legacy of nasa’s operation icebridge,” 2021.
[16] C. Williams and C. Rasmussen, “Gaussian processes for regression,” Advances in neural information processing systems, vol. 8, 1995.
[17] R. Webster and M. A. Oliver, Geostatistics for environmental scientists. John Wiley & Sons, 2007.
[18] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.
[19] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[20] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” arXiv preprint arXiv:1312.6114, 2013.
[21] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

Extra Slides

Future Work

Implement Terrain Roughness Index (TRI) as a Loss or Early Stopping Metric

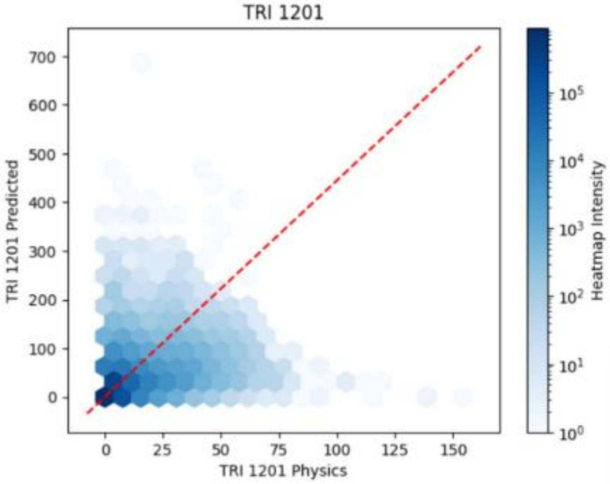
TRI Tested on Additional Validation Data vs. Physics Model

Physics Target, TRI Mean: 3.302 (physics model is over-smooth)

Predicted Data, TRI Mean: [14.178-15.926]

Why implement it as early stopping?

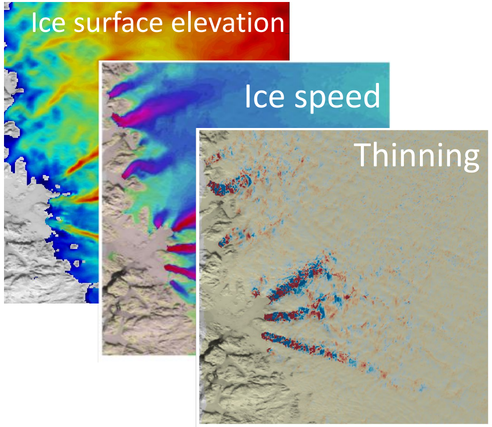
Reduce extreme predictions in flatter areas since our data is rigorous terrain



Data Sources

- Surface variables
    a. Surface elevation (height, m) from Greenland Ice Mapping Project (GIMP/GrIMP) (2012)
    b. Rate of ice thinning (m/yr) from ICESat-2
    c. Surface mass balance (m) from RACMO.
    d. Ice flow surface velocity (m/yr) from satellite interferometry

- Target variable: ice bed elevation
    a. 632,706 data points were collected via ice penetrating radar, from NASA IceBridge mission



Interpolation

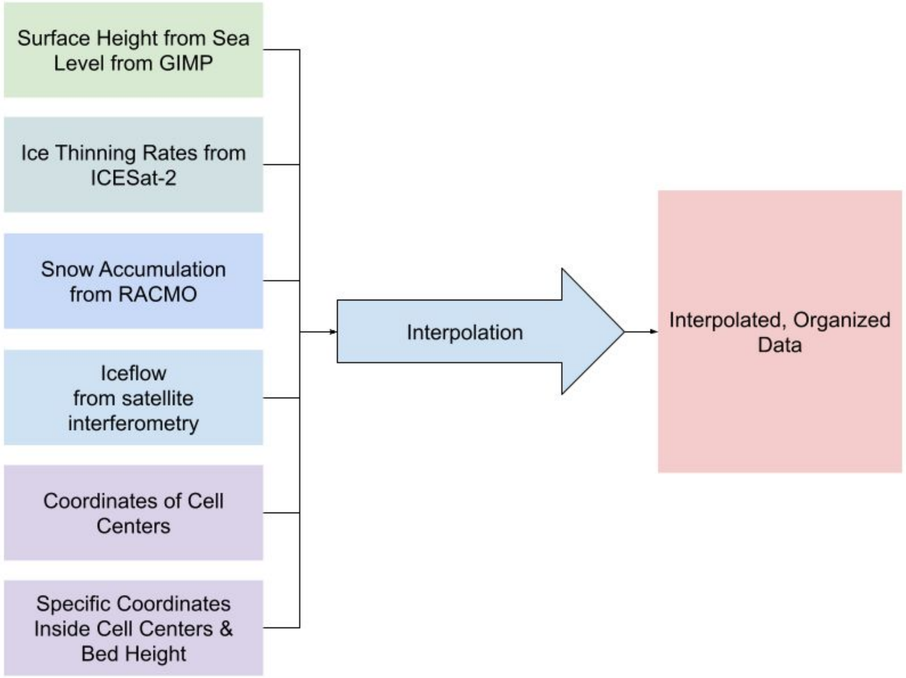
Why interpolation is necessary

Why interpolation?

- The 5 ice surface variables/features are in 2D grid structure with 150 meter spatial resolution
- The ground-truth data (ice bed elevation) is track data and was collected with various distances between two adjacent track pixels (75-150 m)
- To train a model, we need to know the estimated feature values for each location of track pixel

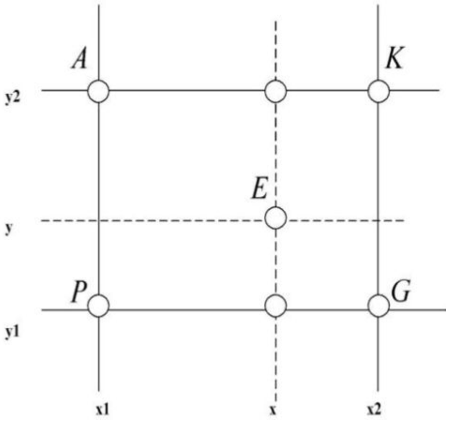
Interpolation approaches explored

- Nearest Neighbor
- Bilinear interpolation
- Universal Kriging



Nearest Neighbor Interpolation

- Nearest neighbor interpolation involves estimating the value of a function at a point where the value is not explicitly known, using the information from nearby points where the function's value is given
- The nearest neighbor algorithm tackles this task by selecting the value of the closest known point without taking into account the values of other nearby points. As a result, it creates a piecewise-constant approximation

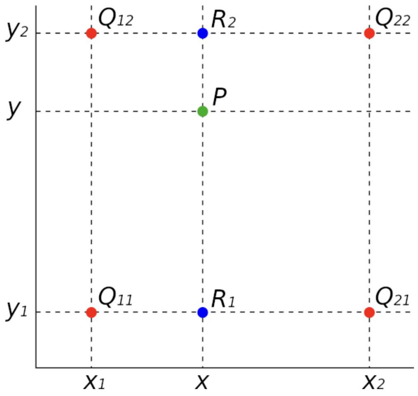


- Each grid is separated by 150 m
- For each track data, the nearest surf_x, surf_y and other feature points are selected using the scaling factor of 150 m

Ref:Rukundo, Olivier and Hanqiang Cao. “Nearest Neighbor Value Interpolation.” *ArXiv* abs/1211.1768 (2012): n. pag.

Bilinear Interpolation

- Bilinear interpolation is a commonly used image resizing and interpolation technique in computer graphics and digital image processing
- It is an improvement over the nearest neighbor interpolation method and provides smoother and more accurate results. In bilinear interpolation, it considers the values of four nearest neighboring pixels from the original image to compute the value of a pixel in the new image
- The method takes the weighted average of these four pixels to estimate the value of the target pixel



- We took the surf_x and surf_y values along with other other features on the grid
- Then we interpolate those values for the track bed data

Ref:https://x-engineer.org/bilinear-interpolation/

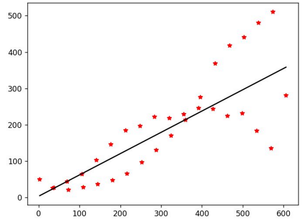
Kriging based Interpolation

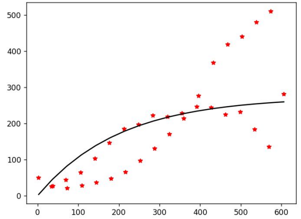
- Spatial probabilistic interpolation algorithm
  - Assumes value of target at position s1 can be predicted as a linear combination of the value of target at all other known positions
    - We use a fitted model called a variogram to determine these weights
- Predict value of each surface feature (surf_vx, surf_vy, etc) at coordinates from radar data (track_bed_x, track_bed_y)

Implementation notes

- dynamically choose best variogram fit using error metrics

High memory cost necessitates some sort of batch processing
-> Draw a box around each bed point, use surface points within that box to fit variogram and make predictions

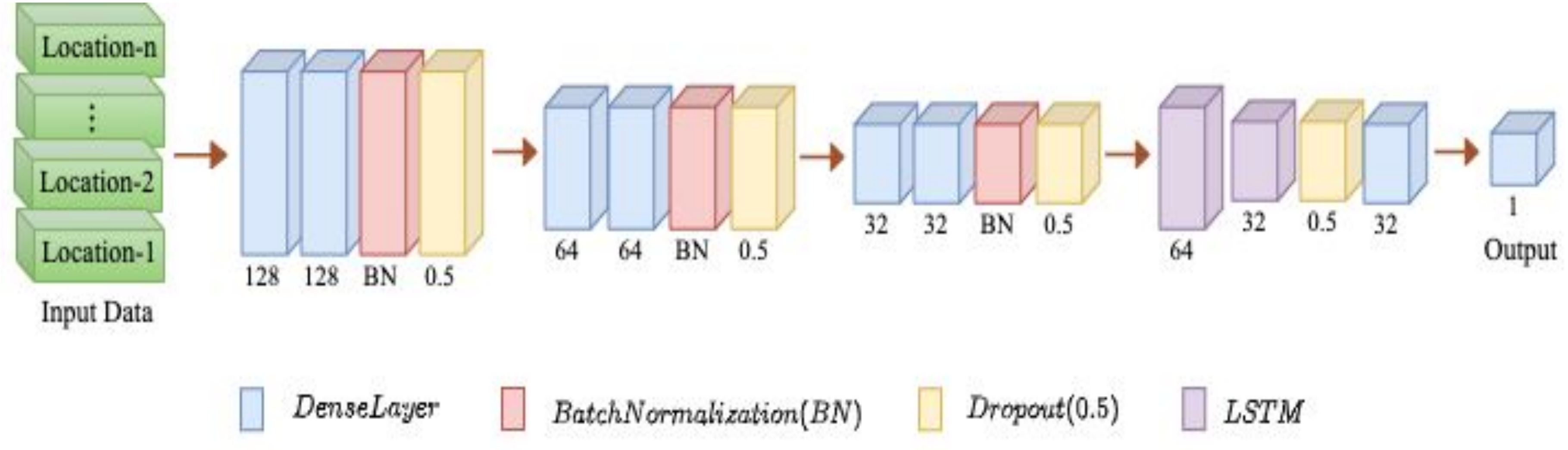




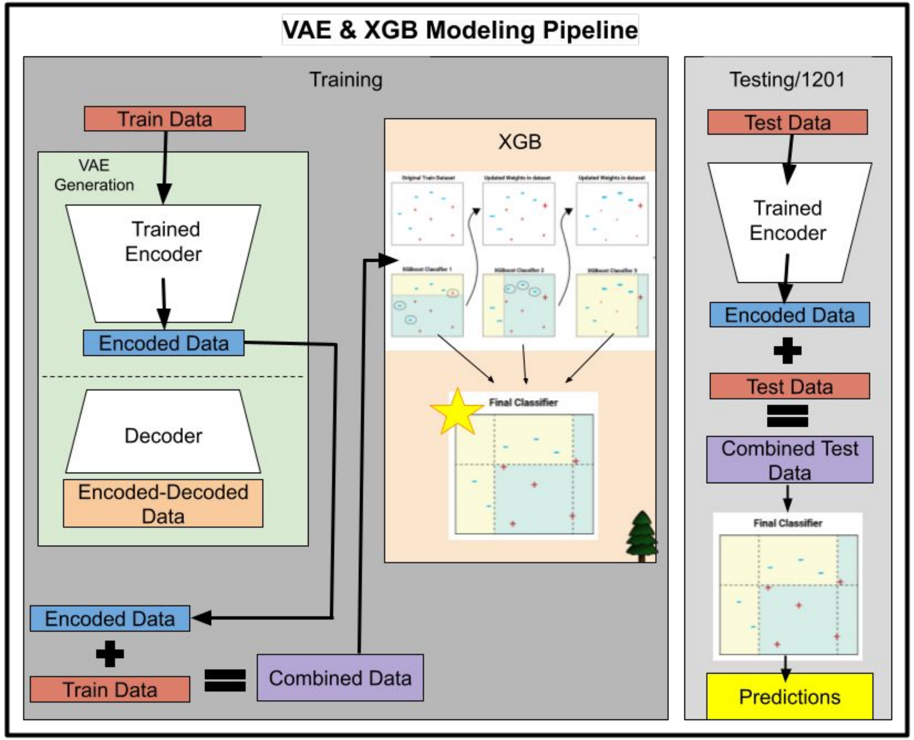
Example of variogram fitting visualized.
x-axis = distance, y-axis = variance from point distance is measured from

Modeling and Metric Validation

Hybrid Methods: Dense+LSTM, Notes



Hybrid Methods: VAE+XGB, Notes

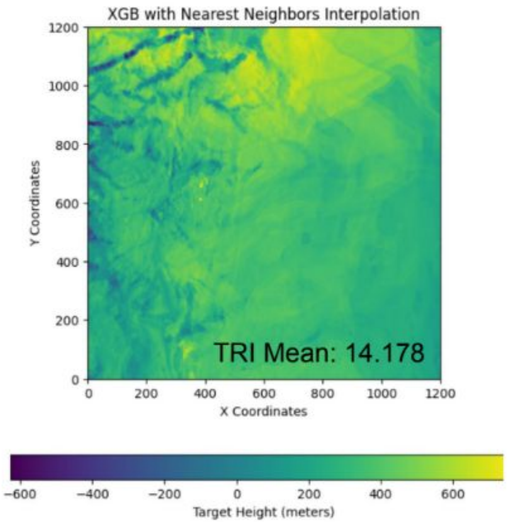


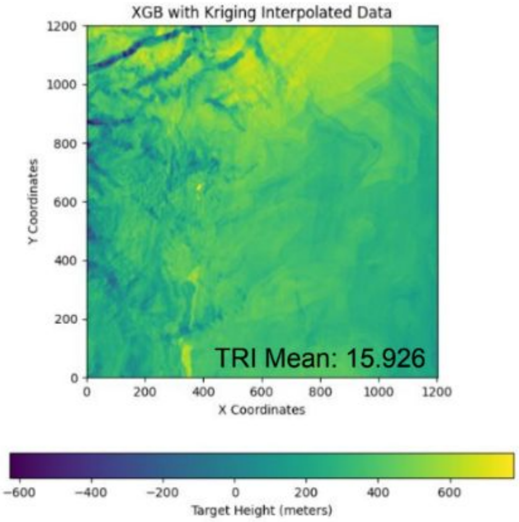
Kriging Residual Learning

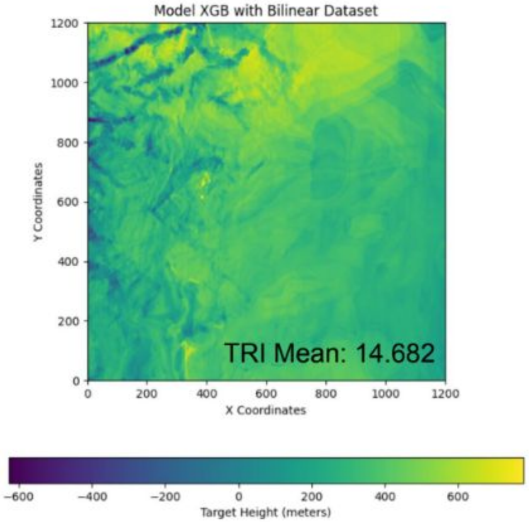
Goal: Learn the error associated with Kriging prediction and correct it.

1. Universal Kriging uses spatial correlation to make a first guess prediction for ice bed elevation (target)
2. Residual (prediction error) is calculated for each prediction where ground truth is known
3. Machine Learning model is trained to predict residual using all features and the kriging prediction
4. Predict all residuals for the given data
5. Take the difference between predicted residuals to the kriging first guess prediction to yield better prediction.

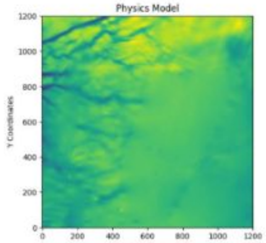
Applying Trained Models for Additional Validation Data







Physics Model TRI Mean: 3.302
(known to be too smooth) →



Result Verification with Terrain Roughness Index (TRI)

Background: Ranges from 0-280+ meters world wide. Lower score is flatter terrain and higher score is more rigorous terrain.
Nunn and Puga (2012) report Greenland as 41m TRI in 1996.
TRI calculation for a single cell:

1. Calculate difference between the center cell and surrounding 8 cells.
2. Average the squared differences.
3. Finally, take the square root.

Results on Validation Data

Metric/interpolation prediction	Known	Kriging	Nearest Neighbors	Bilinear
Mean	173.404	**169.498**	166.093	168.384
R^2 vs. known	-	**0.951**	0.932	0.950
RMSE vs. known	-	**24.635**	29.151	24.847
MAE vs. known	-	**17.101**	20.859	17.326