From Data to Topography: Deep Learning Approach to Predict Ice Bed

Omar Faruque, Homayra Alam, Emam Hossain, Md Alomgeer Hussein

Final Report, Deep Learning (IS 757)
Department of Information Systems, UMBC
{*omarfaruque, halam3, emamh1, mdalomh1*}@*umbc.edu*

What: Estimating topography (bedrock beneath the ice) of Greenland Ice bed
Why: CONTROL the flow of ice, subglacial drainage, and impact of climate change as ice sheets melt -- (& help inform decisions on policy to protect icesheets)

Abstract

How: Combining all features of ice bed surface and using dense and RNN layers to predict future patterns

Greenland bed topography is important to estimate to control the flow of ice, subglacial drainage, and the impact of climate change on the ice sheet. A lot of information about the bed is transmitted to the surface of the ice but the amount of the satellite data is pretty low compared to the huge ice sheets. Therefore, those missing data points need to be predicted by combining all the features of the ice bed surface of Greenland. Here, we have presented a deep learning-based approach to predict the ice bed topography. It is hard for the machine learning model to capture the hidden pattern of the features. A combination of dense and LSTM layers is able to predict the complex pattern of the dataset. Some traditional machine learning models are applied to predict the ice bed map but the deep neural network seemed promising.

Introduction

Sea levels are rising as a result of the melting of the Greenland Ice Sheet (GrIS). Computer models are used to precisely anticipate how these ice sheets will be affected by climate change. The structure of the bedrock beneath the ice, however, has a significant impact on how reliable these models are. Thus, having a precise depiction of the bed topography is essential. Due to scarce measurements, bed topography data are currently incomplete, hence it is important to build methods for addressing these data gaps.

why topography

The shape of the bedrock buried under thousands of meters of ice was the key to future sea level rise. Since the ice has retreated in Greenland, glacier fjords are now exposed. but it's very difficult to see through the ice and measure what the stands cape looks like. Scientists have been making computer models in order to make predictions of sea level rise and how ice sheets are affected by climate change. They have been doing this for so many years while it is proved that though today we can measure the sea level rising, we don't know whether the ice sheets are going to catastrophically collapse over the next 100-200 years. It's just very difficult to do since there is a lot of uncertainty in sea level predictions and it remains too uncertain for policymakers to make informed decisions about how to cope with rising seas to understand how the ice sheets respond to climate change. Due to the cold weather in Greenland, most of the snow that falls every year on the ice sheet doesn't melt away and so year after year this snow slowly becomes solid ice. Ice sheets and glaciers are big pieces of ice cubes that don't move but they actually deform under their own weight like a very viscous fluid.

why important con't

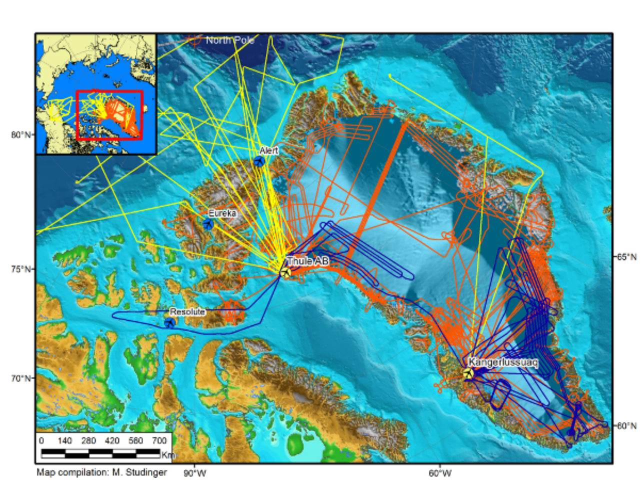


Figure 1: GrIS ice bed map grids: each grid is 150m apart. The red and blue lines show which points are already known. (Morlighem et al. 2020)

Over the past two and three decades, glaciers are putting more icebergs into the ocean and also in the Arctic in Greenland. The rate at which the ice sheets are losing mass depends to a large extent on the shape of the bedrock. There are two reasons for that. The first one is warm water in the ocean that stayed away from the ice previously. It used to not interact with the ocean. But because of climate change, warm water intrusion into fjords under the ice. There is flowing ice over there and it is melting from below. So, if the bed is shallow and if it has big ridges mountain wrenches it may block this warm water at depths under 300-400 meters. The glaciers that have a big bump in front of them may be protected. But the shape of the bedrock can make a difference between a slow gradual retreat or a fast unstoppable one. This is because the ice sheets weigh a lot of mass and over time it has pushed the land below. In the interior of the ice sheet where the bed is below sea level is called *retrograde bed slope*. Here the bed gets deeper as moving towards inland and it isn't stable. When a glacier starts to retreat in

I only read one reason -- warm water intrusion.

confusing

a retrograde region where the bed gets deeper and deeper there is no stopping, it will continue to do that. Therefore, it is needed to know where the bed is retrograde and where it has bumps and ridges that may stop that retreat. If a good representation of the bed is not available, there is no way to make accurate predictions of sea level rise. Since it's incredibly difficult to see through thick eyes how the bed map looks, the information of what lying under the ice bed is verily needed to better under climate change.

This project's objective is to predict the missing data points in Greenland's bed topography using data that has been communicated to the ice's surface. The available datasets from satellite/radar imagery, such as ice flow speed, surface height, and ice thinning rates, will be combined to create a complete view of the ice bed topography. Machine learning and deep learning models will be used to predict these missing points of ice bed topography. In general, the goal is to enhance the prediction of the effect of climate change on ice sheets and to develop a thorough understanding of the Greenland ice bed topography.

Related Works

Due to the effects on sea level rise, ice dynamics, and ocean circulation, research into predicting ice bed topography is a crucial field in climate science (Morlighem et al. 2014; Fretwell et al. 2013; Morlighem et al. 2017). Important contributions include the work of (Morlighem et al. 2014), who used the shallow ice approximation and mass conservation principles to determine the bed topography beneath the Greenland Ice Sheet. Similar to this, (Fretwell et al. 2013) created the Bedmap2 project, which offers a thorough topographic data model of the Antarctic ice bed.

Gravity measurements were used to acquire data on bed topography and ice thickness until the early 1970s when they were replaced by aerial radar sounders (Bentley 1972; Dowdeswell and Evans 2004). Our understanding of the topography of the Greenland ice sheet bed has recently been transformed by NASA's (MacGregor et al. 2021), which has recently greatly enhanced the ice thickness data.

Additionally, (Le Brocq et al. 2013) applied machine learning to this area by forecasting Antarctic bed topography using artificial neural networks. Deep learning techniques' potential applications, however, have not been fully investigated.

In order to fill in any gaps in our understanding of the bed topography of Greenland, our effort is a continuation of earlier research that makes use of both machine learning and deep learning. Gradient Boosting (Friedman 2001), XGBoost (Chen and Guestrin 2016), and Lasso Regressor (Tibshirani 1996) were among the machine learning models we used. These models were chosen for their demonstrated success in a variety of regression and classification tasks, feature selection, and time series data analysis.

Furthermore, we used deep learning models such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNN) (Hochreiter and Schmidhuber 1997) and Long Short-Term Memory (CNN)(LeCun, Bengio, and Hinton 2015). The handling of any temporal relationships in our data was made easier by the LSTM's capacity to learn and remember over lengthy sequences. The Adam optimizer from (Kingma and Ba 2014) was also applied due to its efficient learning rate modifications.

duplicates in the list

Therefore, our initiative builds on previous work in these fields and applies it to the problem of ice bed topography prediction, which advances our knowledge of climate science.

Dataset

Data Collection

The dataset collected here is basically based on radar. So, a big radar is mounted under the wings of an airplane and it emits a signal that will penetrate the ice. But we only get information directly underneath the aircraft. In figure 1, all these colorful lines are the lines for which we have measurements of the bed so this represents 50 years of international campaigns. These ice sheets are huge and there are many places where we don't have any measurements within a radius of tens of hundreds of kilometers. The task is to predict those missing data points to get a full idea of the complete ice bed map. Table 1 summarizes the available features of the dataset.

Slightly informal; ok information.

Table 1: Description of the Dataset

Feature Name	Feature Description	Unit
surf_x, surf_y	Coordinates of cell centers	m
surf_vx, surf_vy	Ice flow velocity vectors	m/yr
surf_elv	Ice surface elevation	m
surf_dhdt	Ice thinning rates	m/yr
surf_SMB	Snow accumulation	m/yr
track_bed_x, track_bed_y	Coordinate of tracking bed points	m
track_bed_target	Height of the ice bed	m

Data Preprocessing

More data background

The dataset contains a total of ten features. Here, the *track_bed_target* is the target feature. Each feature is a 2D matrix of 1201 rows and 1201 columns. Therefore, each feature has 1201×1201=1,442,401 points. Among them, we know about 396,734 points for training and 235,972 points for testing as track bed data. So, our known data is (396,734+235,972)= 632,706 points which are about 43.86% data of a total of 1.4M points.

Each data grid is separated by 150m. So for each track bed point, the corresponding data is calculated for other features. We took the first point of *surf_x* and subtract it from all the points of *track_bed_x* and divide it by 150. So, (*track_bed_x* - *surf_x[0,1]*)/150 = first element of the index (p). Similarly, the same procedure is done for the last element of *surf_y* for the value of the *track_bed_y*. (*surf_y[-1,0]* - *track_bed_y*)/150 = second element of the index (q). According to that index value (p,q), other feature values are mapped for (*surf_smb*, *surf_vx*, *surf_vy*, etc) and the dataset is created.

what is (p,q)?

What exactly are these features?

Methodology

This section describes the methods we used in this study for *track_bed_target* prediction. Each model has used seven input features and the *track_bed_target* is the output feature. We have used $396,734$ samples for training data and $235,972$ samples for testing. The training data is standardized before feeding into the model. Both traditional machine learning models and deep learning models are used to predict the track bed target points. The results were evaluated based on RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), and coefficient of correlation (R^2 score).

Machine Learning Models

We tried the Gradient Boosting regressor, Extreme Gradient Boosting (XGBoost) regressor, and Lasso regressor as traditional machine learning models to predict target ice bed height. These models are described in the following subsections.

Gradient Boosting Regressor Boosting algorithm is a branch of the ensembled machine learning method, which effectively controls bias-variance tradeoffs. Boosting applies a set of weak predictors on the dataset and provides better accuracy by combining these learners. Gradient boosting (Natekin and Knoll 2013) is one of the most popular ensemble algorithms, powerful enough to find any nonlinear relationship between the target value and independent input features. For the gradient boosting method, we have to define a loss function. In the ensemble step, this method applies a base regression tree and computes the error in prediction, then builds subsequent trees for correcting errors of previous trees. After generating a sufficient number of regression trees, the method aggregates generated trees one at a time and follows the gradient descent method to minimize the loss function when adding trees. We analyzed different hyperparameters to generate the best possible prediction by the gradient boosting method. For our dataset, the gradient boosting method provided the best prediction with 64 weak estimators, where the maximum depth of each estimator is set to 4, the minimum number of samples required to split an internal node is 8, the minimum samples in a leaf are 7, and the number of features to consider when looking for the best split is 8.

Extreme Gradient Boosting Regressor Extreme Gradient Boosting or XGBoost (Chen and Guestrin 2016; Akande et al. 2022) for short is an open-source modular tree-boosting machine learning framework. It's an optimized version of the gradient-boosting algorithm with parallelization. Parallelizing the whole boosting process greatly improves the training time. In traditional methods, the best possible model is trained on the whole training dataset. Instead, XGBoost trains thousands of models on various subsets of the training dataset and then votes for the best-performing model. Like other gradient boosting methods, XGBoost tires to minimize the loss gradient by combining best-performing regression trees. Furthermore, XGBoost minimizes a regularized (L1 and L2) objective function that combines a loss function based on the difference between the predicted and target outputs and a penalty term for model complexity. The parallel processing ability of XGBoost is the most important factor for its success in various situations. On a single processor, the technology is ten times faster than other common implementations, and it scales to billions of examples in distributed or memory-limited environments. Similar to the gradient boosting method, the XGBoost regressor is also trained using 64 estimators with the maximum depth of a tree estimator being 4 where 80% of the training data is used to train each tree regressor. From all independent variables, 50% of the columns will be used as features to calculate the best split, and the learning rate is set to 0.1.

Lasso Regressor Lasso regression (Tibshirani 1996) is a linear regression method that uses L1 regularization to impose a penalty on the sum of the absolute values of the coefficients. This penalty encourages the model to select a smaller set of relevant features and sets the coefficients of irrelevant features to zero. The resulting model is easier to interpret and less prone to overfitting, making it particularly useful when dealing with high-dimensional data. The regularization strength parameter, which controls the trade-off between the fit to the training data and the magnitude of the coefficients, can be selected through cross-validation. For our target task the hyperparameters of the Lasso regressor, such as the regularization strength (*alpha*), whether to fit the intercept (*fit_intercept*) and the maximum number of iterations (*max_iter*) were specified to be optimized using GridSearchCV, which exhaustively searches over the specified hyperparameters and returns the best combination based on the negative mean squared error (MSE) score. The GridSearchCV method demonstrates that the best hyperparameters found by the optimization were *alpha*=0.0001, *fit_intercept*=True, and *max_iter*=500, which resulted in an MSE score of 0.6118. Then Lasso regressor with these best hyperparameters is used to test the performance of the model on the test dataset.

Deep Learning Models

Deep learning models are widely used in both classification and regression applications. Deep learning models are very good at learning features from large datasets and predicting values of target variables from any complex system. To find the inherent relationship among the dependent variables and their contribution to predicting the target ice bed height we applied different deep learning models to our dataset. Applied models are described in subsequent subsections.

Dense Layer-based Regression The dense layers are the most common layers in deep neural networks and their operation is very simplistic. Each neuron of a dense layer takes input from every neuron of the preceding layer and aggregates their result in a nonlinear manner. To create the model we used three dense layers and each pair of dense layers is separated by a dropout layer. Finally, the prediction result is taken from the last layer of the model with only a single neuron. The model architecture is given in figure 2. The ReLU activation function is used to get nonlinear activation results from dense layers. The model is trained using the Adam optimizer with the mean squared error objective function. The

Adam optimizer is very efficient to find the global optimum of the model weights as it utilizes the momentum of the error gradient of the objective function. The goal is to reduce the error between the predicted ice bed height and the actual ice bed height in the training dataset. During the training phase, the model is validated using 40% of the testing data with random shuffling. As the ground truth value of the ice bed height of training data has a high variance, we used a large batch size of 20,000 and trained the model for 100 epochs. To generalize the learned weights, 50% dropout is used in each of the dropout layers.

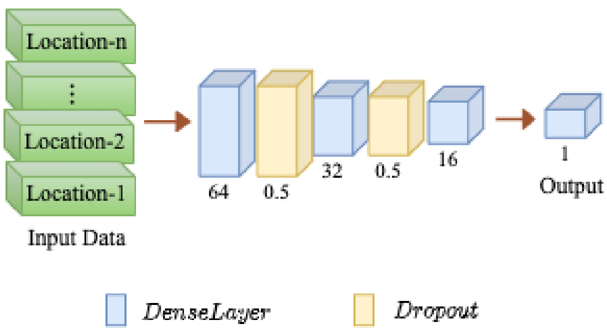


Figure 2: Visualization of the Dense Layer based model.

LSTM Regression Long-Short-Term Memory (LSTM) is a special type of recurrent neural network, that can easily capture available patterns from sequential data. Normal recurrent neural networks face the problem of vanishing gradient during the back-propagation of the objective function. But LSTM cells have long-term and short-term memory that helps to preserve information from previous distant sequences and also to prevent the vanishing gradient issue. We used LSTM for predicting the ice bed height measurement from the dependent variables. To apply the LSTM model we have transformed the dataset from (index, variables) to the shape of (index, sequence, variables). Three LSTM layers are used in this model with 64, 32, and 16 cells in each subsequent layer. Between each pair of LSTM layers, we used a dropout layer to increase the generalizability of our model. The output layer of the model contains a single neuron with linear activation to generate the prediction result. The model design is illustrated in figure 3. The optimization of the LSTM regression model is carried out using the Adam optimizer and mean squared error is used as the objective function. Similar to the dense layer model, this model is likewise trained using a batch size of 20,000 and 100 iterations.

*** best

Dense Layer and LSTM Regression Integrating dense layer and LSTM layers within a model helps to generate latent features helpful to detect hidden patterns from the datasets. In the current regression task, the dataset has only 5 independent variables in each grid location to predict the target value and also their interaction is very complex. So we used this integrated dense layer and LSTM model to generate latent features for this complex system. This model contains three dense layer blocks containing two dense layers, one batch normalization layer, and one dropout layer with a 50% dropout coefficient. The sigmoid activation function is used in each of these dense layers. The LSTM block of this model is constructed using two LSTM layers of 64 and 32 cells, then a batch normalization layer and one dense layer. The result of this model is generated using a single neuron with the linear activation function. Similar to the LSTM regressor model, the dataset is transformed from a 2D array to a 3D array with sequence parameters to apply this integrated model. Two hyperparameters of this model are batch size 20,000 and epochs 100. This model is optimized using the Adam method and mean squared error is used as the objective function. In each epoch of the optimization process, 60% of the training data is used for training, and 40% of the training data is used to validate the model performance. Figure 4 shows the architecture of the model.

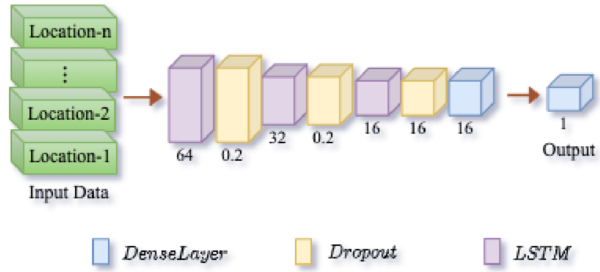


Figure 3: Visualization of the LSTM model.

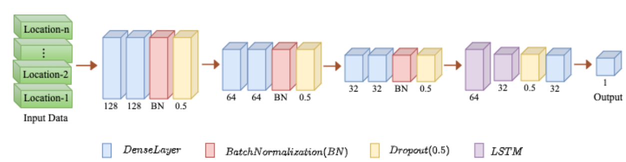


Figure 4: Visualization of the integrated Dense Layer and LSTM model.

Results and Discussion

To evaluate the performance of the models, we considered RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), and R^2 (coefficient of correlation). The quantitative evaluation of the generated regression results of the proposed model against the five baseline models is represented in Table 2. Among the baseline models, XGBoost performs better than the others. However, the LSTM model combined with dense layers (Dense+LSTM) significantly performs better than all other methods. Compared to Dense+LSTM, the second best model XGBoost has 136% higher RMSE, 76% more MAE, and 70.5% lower R^2 score.

Figures 5, 6, and 7 illustrate the prediction results of the Gradient boosting, XGBoost regressor, and Dense+LSTM model where the color code represents the height of the track bed.

Table 2: Performance Comparison

Model	RMSE	MAE	R^2
Gradient Boosting	136.93	100.63	0.113
XGBoost	135.18	97.06	0.135
Lasso Regressor	147.30	106.79	-0.026
Dense	387.98	364.02	-6.110
LSTM	556.09	493.71	-13.620
Dense+LSTM	**57.30**	**55.04**	**0.840**

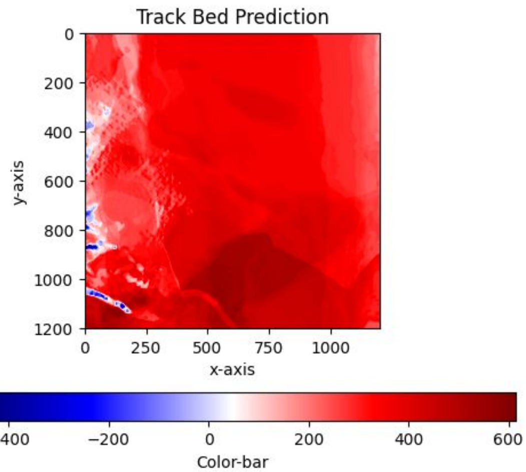


Figure 5: Track bed prediction map using Gradient Boosting method on 1201 × 1201 grid.

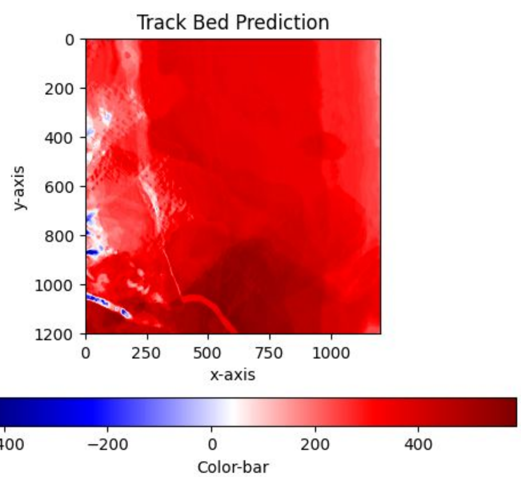
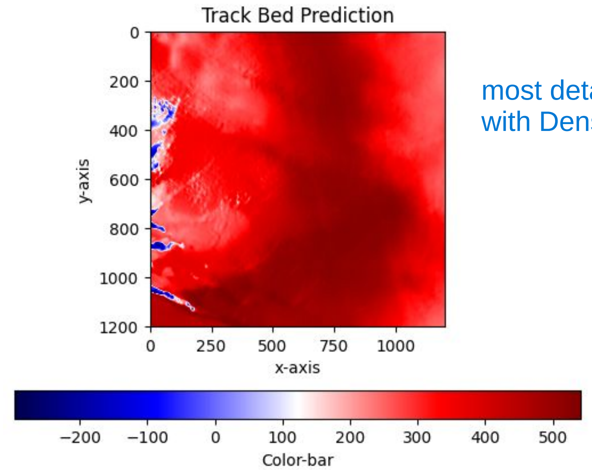


Figure 6: Track bed prediction map using XGBoost method on 1201 × 1201 grid.

Conclusion & Future Works

In this study, we investigated various machine learning and deep learning models for predicting track bed target points using seven input features. We tested Gradient Boosting, XGBoost, Lasso Regression, Dense Regressor, LSTM, and a combination of Dense+LSTM models with different con-



most detail seen
with Dense + LSTM

Figure 7: Track bed prediction map using Dense+LSTM based Deep-learning model on 1201 × 1201 grid.

figurations. By evaluating the models using RMSE, MAE, and R^2 scores, we aimed to identify the most effective approach for predicting missing data points in ice bed topography. This work contributes to a better understanding of the Greenland Ice Sheet and ultimately helps improve climate change predictions and adaptation strategies. The dataset has multiple track bed heights for the same grid point. This can pose a challenge for ML/DL models since they have to learn different target values for the same coordinates. To solve this problem, future work will be to remove the duplicate data points using different interpolation methods.

concl

future

References

Akande, Y. F.; Idowu, J.; Misra, A.; Misra, S.; Akande, O. N.; and Ahuja, R. 2022. Application of XGBoost Algorithm for Sales Forecasting Using Walmart Dataset. 147–159.

Bentley, C. R. 1972. Seismic-wave velocities in anisotropic ice: A comparison of measured and calculated values in and around the deep drill hole at Byrd Station, Antarctica. *Journal of Geophysical Research*, 77(23): 4406–4420.

Chen, T.; and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. 785–794.

Dowdeswell, J. A.; and Evans, S. 2004. Investigations of the form and flow of ice sheets and glaciers using radio-echo sounding. *Reports on Progress in Physics*, 67(10): 1821.

Fretwell, P.; Pritchard, H. D.; Vaughan, D. G.; Bamber, J. L.; Barrand, N. E.; Bell, R.; Bianchi, C.; Bingham, R.; Blankenship, D. D.; Casassa, G.; et al. 2013. Bedmap2: improved ice bed, surface and thickness datasets for Antarctica. *The cryosphere*, 7(1): 375–393.

Friedman, J. H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189–1232.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Le Brocq, A. M.; Ross, N.; Griggs, J. A.; Bingham, R. G.; Corr, H. F.; Ferraccioli, F.; Jenkins, A.; Jordan, T. A.; Payne, A. J.; Rippin, D. M.; et al. 2013. Evidence from ice shelves for channelized meltwater flow beneath the Antarctic Ice Sheet. *Nature Geoscience*, 6(11): 945–948.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *nature*, 521(7553): 436–444.

MacGregor, J. A.; Boisvert, L. N.; Medley, B.; Petty, A. A.; Harbeck, J. P.; Bell, R. E.; Blair, J. B.; Blanchard-Wrigglesworth, E.; Buckley, E. M.; Christoffersen, M. S.; et al. 2021. The scientific legacy of NASA's Operation Ice-Bridge.

Morlighem, M.; Rignot, E.; Binder, T.; Blankenship, D.; Drews, R.; Eagles, G.; Eisen, O.; Ferraccioli, F.; Forsberg, R.; Fretwell, P.; et al. 2020. Deep glacial troughs and stabilizing ridges unveiled beneath the margins of the Antarctic ice sheet. *Nature Geoscience*, 13(2): 132–137.

Morlighem, M.; Rignot, E.; Mouginot, J.; Seroussi, H.; and Larour, E. 2014. Deeply incised submarine glacial valleys beneath the Greenland ice sheet. *Nature Geoscience*, 7(6): 418–422.

Morlighem, M.; Williams, C. N.; Rignot, E.; An, L.; Arndt, J. E.; Bamber, J. L.; Catania, G.; Chauché, N.; Dowdeswell, J. A.; Dorschel, B.; et al. 2017. BedMachine v3: Complete bed topography and ocean bathymetry mapping of Greenland from multibeam echo sounding combined with mass conservation. *Geophysical research letters*, 44(21): 11–051.

Natekin, A.; and Knoll, A. 2013. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7: 21.

Tibshirani, R. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1): 267–288.